

AD-A147 211

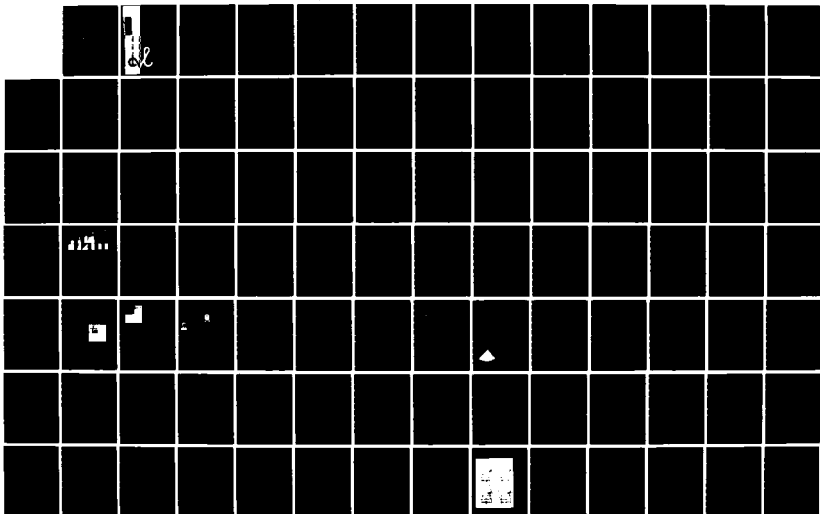
METRIC ASPECTS OF DIGITAL IMAGES AND DIGITAL IMAGE  
PROCESSING(U) PURDUE UNIV LAFAYETTE IN SCHOOL OF CIVIL  
ENGINEERING E M MIKHAIL ET AL. SEP 84 CE-PH-84-9  
ARO-17508. 8-GS DAAG29-81-K-0063

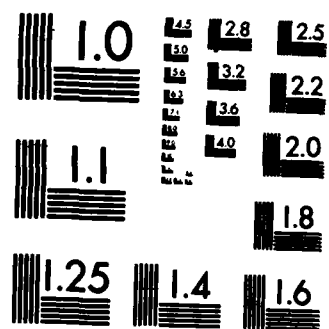
1/2

UNCLASSIFIED

F/G 8/2

NL





# SCHOOL OF CIVIL ENGINEERING

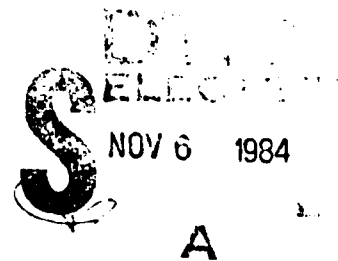
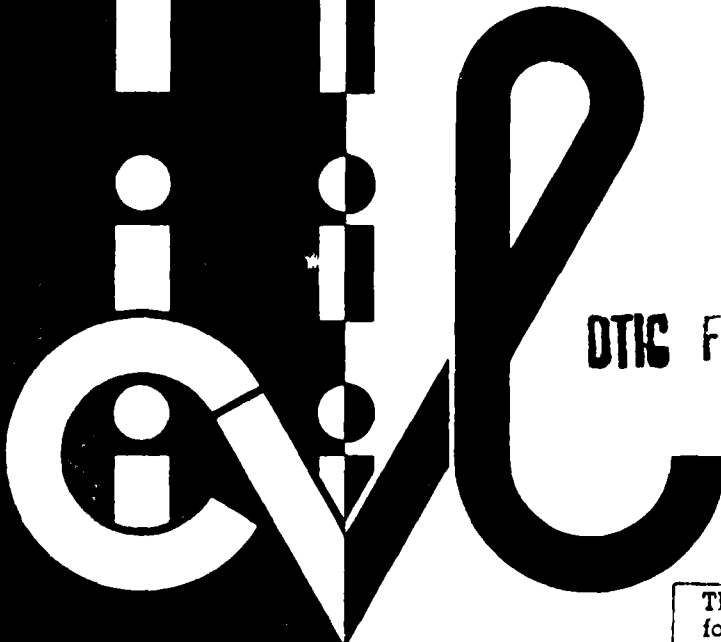
AD-A147 211

SURVEYING, MAPPING, GEODESY,  
AND PHOTOGRAMMETRY

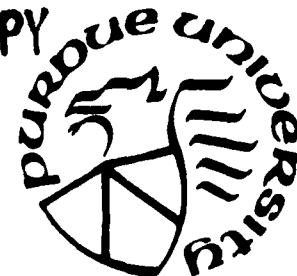
PH-84-9 September 1984

METRIC ASPECTS OF DIGITAL IMAGES  
AND DIGITAL IMAGE PROCESSING

E. M. Mikhail and G. R. Mitchell



DTIC FILE COPY



This document has been approved  
for public release and sale; its  
distribution is unlimited.

PURDUE UNIVERSITY

84 11 01 103

**SURVEYING, MAPPING, GEODESY,  
AND PHOTOGRAMMETRY**

**CE-PH-84-9      September 1984**

**METRIC ASPECTS OF DIGITAL IMAGES  
AND DIGITAL IMAGE PROCESSING**

**E. M. Mikhail and O. R. Mitchell**

**DTIC  
ELECTE  
NOV 8 1984  
S A D**

**CE-PH-84-9**

**Metric Aspects of Digital Images and Digital Image Processing**

**Final Report**

**by**

**Edward M. Mikhail**  
Principal Investigator  
School of Civil Engineering  
Purdue University

**and**

**O. Robert Mitchell**  
Faculty Associate  
School of Electrical Engineering  
Purdue University

**September 1984**

**for**

**United States Army Research Office**

**Contract Number**

**DAAG29-81-K-0063**

**Approved For Public Release;  
Distribution Unlimited.**

The views, opinions, and/or findings contained in this report are those of the authors and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.

## Table of Contents

	Page
Foreword .....	iv
Chapter 1 - Human Pointing Ability in a Digital Data Base .....	1
1.1 Effects of Sampling .....	1
1.2 Measurements of Hard-Copy Images .....	2
Chapter 2 - Automatic Extraction of Metric Information .....	4
2.1 Precise Location of Edges .....	4
2.1.1 Grey Level Moment Preserving Edge Location .....	4
2.1.2 Edge Location using Least Squares Adjustment .....	6
2.1.3 Empirical Analysis .....	8
2.1.4 Spatial Moment Based Two-Dimensional Edge Operator .....	9
2.2 Precision Cross Target Location .....	11
2.2.1 Pattern Recognition of Crosses .....	11
2.2.2 Location using Moments .....	12
2.2.3 Location using Fourier Descriptors .....	12
2.2.4 Evaluation using Arizona Data Base .....	13
2.2.5 Location using Least Squares .....	14
Chapter 3 - Effects of Image Processing on Geometric Fidelity .....	17
3.1 Effects of Digital Image Compression - Cosine Compression .....	17
3.2 Effects of Mean and Median Filters in the Presence of Noise .....	18
3.3 Effects of Resampling using Various Interpolation Functions .....	20
References .....	23
Appendices .....	24
Appendix A - Photogrammetric Analysis of Digital Images	
Appendix B - Edge Location to Subpixel Accuracy in Digital Imagery	
Appendix C - A Moment Based Two-Dimensional Edge Operator	
Appendix D - Detection and Sub-pixel Location of Photogrammetric Targets in Digital Images	
Appendix E - Detection and Sub-pixel Location of Objects in Digitized Aerial Imagery	
Appendix F - The Circular Median Filter	
Appendix G - Participating Scientific Personnel	
Appendix H - Publications, Technical Reports, and Theses	

## Foreword

This final technical report concisely documents the results of a three-year research effort undertaken by the Schools of Civil and Electrical Engineering at Purdue University. The primary purpose of the work has been to investigate the metric aspects involved with digital images and digital image processing. The emphasis of the research has been on metric fidelity of images which is the main thrust of various photogrammetric tasks, dealing with high accuracy positional information. Study of mensuration of digital hard-copy images, which may have been subjected to digital image processing algorithms, is needed in order to determine the expected accuracies of metric information extracted from such images. This called for a cooperative effort between researchers with specialities in photogrammetry and digital image processing.

The report is divided into three primary areas of research. Not coincidentally, these areas follow the tasks originally proposed for investigation. Quantitative analysis of digital sampling of images is expounded in Chapter 1. Determining metric criteria for evaluation of digital image processing techniques is the research described in Chapter 2. Evaluation of various digital image processing techniques is documented in Chapter 3. More specifically, Chapter 1 describes the relationship between the pixel size and the precision and accuracy with which objects can be located by human observers using digital hard-copy images in photogrammetric plotters. Chapter 2 describes the research involved in developing precision edge and cross target location methods which have subpixel accuracy. Chapter 3 documents the type and amount of metric distortion caused by various digital image techniques such as compression, filtering, and resampling. The amount of distortion is measured by those methods developed in Chapter 2.

The authors would like to thank Dr. S.J. Mock of the Army Research Office for his continued support and guidance throughout the period of the contract. Valuable assistance was provided by Dr. V.L. Anderson of Purdue University, and Dr. J.E. Unruh of the Defense Mapping Agency Aerospace Center who made possible the use of the Optronics film-writing equipment and provided much useful advice.

During the course of this research, the following contributed directly or indirectly to its completion and success: Dr. A.Y. Tabatabai, Dr. J.D. Thurgood, Dr. W. Forstner, Dr. J.S. Bethel, Mr. M.L. Akey, Mr. D.B. Cantiller, Mr. D. Davis, Mr. M. Meenehan, Mr. B. Nickols, Mr. W.A. Oren, Mr. F. Paderes, and Mr. A. Sifaw. The authors express their gratitude to them.



## **1. Human Pointing Ability in a Digital Data Base**

The efforts of this research were directed toward determining how image processing techniques affect the metric quality of digital images. As in most research work concerned with image processing, some type of criteria was established to measure or judge these techniques. Toward this end, the precision of a human observer was established and this precision was used as a standard of comparison. This section details the ability of the observer to accurately point to targets with subpixel results. In addition, the observer consistency as well as inter-observer variability was determined.

### **1.1. Effects of Sampling**

Preliminary work has shown the effects caused by sampling a hard-copy image to be minimal. This work which was performed by Mikhail and Unruh [1] quantified the measurement accuracy in human pointing ability on sampled digital images. In general, degradation of the metric accuracy can be attributed to the artifacts produced in a reconstructed digital image.

Synthesized aerial photographs were formed by processing a combined elevation and orthophoto data base. These simulated photographs were digitized and written on film with pixel sizes of 25, 50, and 100  $\mu\text{m}$ . Since the image geometry was completely controlled, image coordinates of specific targets were calculated and compared with measured quantities. In addition, two types of targets were used. The first type appeared as crosses in the data base prior to synthesis and each cross covered many pixels in the digital image. The second type simulated an artificial image point (such as a PUG mark) and was constructed by darkening a single picture element (pixel) in the digital image.

Monoscopic measurements of single pixel targets resulted in a precision of 7  $\mu\text{m}$  regardless of pixel size (Table 1). For the 25 and 50  $\mu\text{m}$  pixel spot sizes, monoscopic measurements exhibited precisions of 8  $\mu\text{m}$  for multiple pixel targets. At 100  $\mu\text{m}$  pixel size, precision dropped to 17  $\mu\text{m}$ . Inference can be made that as the pixel spot size was increased, the human observer's precision degraded. However, this occurred only when the pixel was large enough to become apparent to the observer. In these instances, the targets appeared "blocky" to the observer.

At 25 and 50  $\mu\text{m}$  pixel sizes, stereoscopic measurements of single pixel targets showed a decrease in precision to 16  $\mu\text{m}$ . However, for the cross targets at the same pixel sizes, precision remained at 9  $\mu\text{m}$ . This indicated the importance of the background features to the observer. Precision of measurement for single pixel targets was dependent on features close to the target. For the 100  $\mu\text{m}$  pixel size, the error increased to 98  $\mu\text{m}$  for the single pixel targets.

Precision and Accuracy for Monoscopic and Stereoscopic Measurements								
	Single Pixel Targets				"Cross" Targets			
	Precision		Accuracy		Precision		Accuracy	
Monoscopic	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )
100 $\mu\text{m}$ pixels	7.5	3.9	9.1	6.5	14.2	11.9	17.7	16.4
50 $\mu\text{m}$ pixels	7.2	3.2	7.9	4.4	9.4	6.8	10.4	7.0
25 $\mu\text{m}$ pixels	5.9	3.7	13.8	4.9	5.1	4.7	12.3	4.7
Stereoscopic	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )	x ( $\mu\text{m}$ )	y ( $\mu\text{m}$ )
100 $\mu\text{m}$ pixels	152	91	187	101	32	15	35	20
50 $\mu\text{m}$ pixels	22	23	24	24	11	12	12	12
25 $\mu\text{m}$ pixels	13	18	13	20	6	9	7	10

**Table 1.** Precision and accuracy for monoscopic and stereoscopic transfer measurements of synthetic photographs.

These results indicate that hard-copy digital images can provide measurement precisions of 7  $\mu\text{m}$  or better for monoscopic viewing and roughly 10  $\mu\text{m}$  for stereoscopic viewing of multiple pixel targets appearing in both images, even with pixel sizes up to 50  $\mu\text{m}$ . Therefore, given that the pixel spot size is sufficiently small, the observer may treat hard-copy digital images as conventional photographs. In addition, the observer is able to accurately locate targets in digital imagery with subpixel precision as low as 0.14 pixels.

## 1.2. Measurements of Hard-Copy Images

This phase of research concerned itself with the observer's accuracy in edge and line location. This work was completed by Thurgood and Mikhail and described in detail in a technical report [2] and a conference paper (Appendix A). The research documents the effects of such factors as the observer consistency, inter-observer variability, and observer precision in the presence of noise. In addition, these results were retained as a standard of comparison with machine methods.

A simulated test set comprised of 15 image files, was constructed. Each of the 15 files contained 4 targets and one image feature. Each file was 1024 by 1024 pixels and was written using a 25  $\mu\text{m}$  square aperture. The targets simulated fiducial marks consisting of 5 by 5 crosses and the image features were either an edge or a line. Three human observers were used to collect measurements using a stereo comparator operated in a binocular viewing mode.

The precision in measurement of the fiducial marks for observers 1, 2, and 3 were 1.1  $\mu\text{m}$ , 3.4  $\mu\text{m}$ , and 2.2  $\mu\text{m}$ , respectively. These values expressed the 68% confidence

limit of a measurement in x or y. No significant difference was detected between precision in x and y. However, the differences in levels of precision between observers was found to be significant. The error in measurement to the edge within each file was investigated using an analysis of variance model. This allowed the independent evaluation for the four factors: replicate, observer, file, and location. In addition, all edge errors were calculated as the normal distance to the edge. Certain relationships were inferred from the results and were shown to be significantly important:

- \* Single observers were not consistent in edge pointing over time.
- \* There existed large significant differences in measured edge locations between observers.
- \* The differences in measured edge location between observers remained consistent.
- \* The mean error and standard deviation increased as the edge spread was increased.
- \* The mean error remained constant as the contrast in the edge was reduced, however the standard deviation increased.
- \* When noise was added to the edge, edge measurement accuracy was affected, but not in a consistent manner with respect to all observers.
- \* Edge orientations affected the edge measurement when the angle of orientation was increased from  $0^{\circ}$  to  $45^{\circ}$ . The edge measurement moved toward the darker side of the edge when the angle was increased from  $0^{\circ}$  to  $45^{\circ}$ .

## 2. Automatic Extraction of Metric Information

Due to the large variance in the human precision measurements research efforts were directed toward implementing a machine vision system that would have a lower variance with possibly higher precision. Two target types were used for precision location: grey level edges and crosses. Separate algorithms were developed to precisely locate each target type. In addition, competing algorithms were implemented to determine the more robust algorithm.

### 2.1. Precise Location of Edges

Of particular importance to the image processing community is the ability to detect and locate grey level edges. To date, few edge detection/location algorithms with subpixel accuracy have been developed. Research was completed in this area resulting in the development of three different types of precision edge locators. The three algorithms include a grey level moment preserving edge locator, an edge locator which uses least squares adjustment, and a spatial moment edge locator. The first two edge locators, the moment preserving and the least squares adjustment, are described in the following two subsections. Comparison is made between these two operators in the third subsection. The spatial moment operator is described in the last subsection.

#### 2.1.1. Grey Level Moment Preserving Edge Location

Tabatabai and Mitchell have developed an edge location method which determines edge locations to subpixel accuracy. The method uses the grey level moments from a window of data. These moments are matched to the first, second, and third order grey level statistics of an ideal continuous edge. Additionally, the operation is insensitive to multiplicative and additive changes in the grey level data. Documentation of this research [3] is in Appendix B.

An ideal edge can be characterized by two grey level values  $h_1$  and  $h_2$  where  $h_1$  is the lower brightness level and  $h_2$  is the higher brightness level. The first, second, and third order moments ( $m_i$ ) are determined by

$$m_i = \rho_1 h_1^i + \rho_2 h_2^i, i = 1, 2, 3 \quad (1)$$

where  $\rho_1$  and  $\rho_2$  are the probabilities or proportions of the grey values  $h_1$  and  $h_2$  that are present in the data, and are given by

$$\rho_1 + \rho_2 = 1 \quad (2)$$

The parameters  $\rho_1$ ,  $\rho_2$ ,  $h_1$ , and  $h_2$  can be solved in terms of the moments, i.e.,

$$h_1 = m_1 - \sigma \left( \frac{\rho_2}{\rho_1} \right)^{1/2} \quad (3)$$

$$h_2 = m_1 + \sigma \left( \frac{\rho_1}{\rho_2} \right)^{1/2} \quad (4)$$

$$\rho_1 = \frac{1}{2} \left( 1 + \frac{s}{\sqrt{s^2 + 4}} \right) \quad (5)$$

$$\rho_2 = \frac{1}{2} \left( 1 - \frac{s}{\sqrt{s^2 + 4}} \right) \quad (6)$$

where

$$s = \frac{m_3 + 2m_1^3 - 3m_1m_2}{\sigma^3} \quad (7)$$

and

$$\sigma^2 = m_2 - m_1^2 \quad (8)$$

For discrete sampled data,  $x_i$ ,  $i = 1, 2, \dots, n$  the moments may be calculated by the equation

$$\bar{m}_i = \frac{1}{n} \sum_{j=1}^n x_j^i, \quad i = 1, 2, 3 \quad (9)$$

If the ordered data  $x_i$  is either monotonically nondecreasing or nonincreasing, then the edge location is

$$\gamma = k = \rho_1 n \quad (10)$$

where the first data sample is located at  $j = \frac{1}{2}$  (all samples have a spacing of one). In general,  $\rho_1$  is noninteger, thus the location  $\gamma$  is a subpixel measurement. It can be shown that the sample skewness

$$\bar{s} = \frac{1}{n} \sum_{i=1}^n \frac{(x_i - \bar{m}_1)^3}{\bar{\sigma}^3} \quad (11)$$

is equal to the "s" in Eq.(7). The conclusion can be made that the edge location is a function of sample skewness only. Since the sample skewness is invariant to additive and multiplicative changes, the edge location is invariant to the same.

In the presence of additive, zero-mean Gaussian noise, the edge location is biased toward the center of the window. This is verified by empirical methods in addition to being derived theoretically.

The one-dimensional edge operator can be extended to two dimensions by using the grey levels within a circular window. The sample moments are

$$\bar{m}_k = \rho_1 h_1^k + \rho_2 h_2^k, \quad k = 0, 1, 2, 3 \quad (12)$$

where

$$\rho_1 = \frac{A_1}{\pi} \quad (13)$$

and

$$\rho_2 = 1 - \rho_1 \quad (14)$$

$A_1$  is that area of the circle covered by the grey level  $h_1$ . The values of  $\rho_1$ ,  $\rho_2$ ,  $h_1$ , and  $h_2$  can now be obtained from Eq.(3) - Eq.(8). For a circle of unity radius and an arbitrary angle  $0 \leq \beta \leq \pi/2$ , the area  $A$  is given by

$$A = \beta - \frac{1}{2} \sin \beta \quad (15)$$

Combining Eq.(13) and Eq.(15) and allowing

$$\rho = \min(\rho_1, \rho_2) \quad (16)$$

results in

$$\beta - \frac{1}{2} \sin 2\beta = \pi \rho \quad (17)$$

This result is transcendental and a numerical approximation is needed to solve for  $\beta$ . However, since this equation is extremely smooth (derivatives of all orders exist), a simple look-up table method which uses linear interpolation is sufficient to obtain quick and accurate results. The length to the edge is simply

$$\ell = \cos \beta \quad (18)$$

As mentioned before, the length is a subpixel result and is invariant to additive and multiplicative changes in the grey level data.

### 2.1.2. Edge Location using Least Squares Adjustment

A process for edge location using a least squares approach has been developed by Thurgood and Mikhail [4,5] (and Appendix A). This approach results in very precise locations when the initial parameter estimates are good.

Let  $f(s,t)$  represent an ideal picture function. In this case,  $f(s,t)$  is the ideal edge function. If the picture function is image using a linear, spatially invariant system with a normalized point-spread function  $p(s,t)$ , the output of the system is simply the convolution of  $f(s,t)$  and  $p(s,t)$ . Let the random variable  $\ell(s,t)$  be the output at coordinate  $(s,t)$ ,

$$\ell(s,t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\zeta, \nu) p(x-\zeta, t-\nu) d\zeta d\nu$$

$$= f(s,t) * p(s,t) \quad (19)$$

where  $*$  denotes the convolution operation. Given that  $f(s,t)$  can be parameterized, the above equation can be transformed into a single condition equation for the model known as Adjustment by Indirect Observations. The set of equations (one equation for each  $\ell(i,j)$ ) can be solved by forming normal equations. The transform is determined by optimizing the coefficients of the convolution of the ideal picture function and point-spread function. Initial estimations of these coefficients are used and a correction matrix is produced. This is repeated using the correction values until the estimates of the parameters of ideal picture function converge.

Let  $f(s,t)$  be completely characterized by a set of  $u$  parameters  $\underline{x}$  over the region of interest, then Eq.(19) may be written as

$$\ell(s,t) - f(s,t;\underline{x}) * p(s,t) = 0 \quad (20)$$

or for simplicity

$$\ell(s,t) + F(\underline{x}) = 0 \quad (21)$$

where  $F(\underline{x}) = -f(s,t;\underline{x}) * p(s,t)$ . The picture element  $(i,j)$  which is a sample of  $\ell(s,t)$  at  $s = s_i, t = t_j$  may be written in the linearized form of Eq.(22) as

$$\ell_{ij}^0 + v_{ij} + B_{ij} \Delta = -F_{ij}(\underline{x}^0) \quad (22)$$

where  $\ell_{ij}^0$  is the initial estimate for the observation,  $v_{ij}$  is the measurement residual,  $B_{ij}$  is the set of partial derivatives of  $F_{ij}(\underline{x})$  with respect to the parameters, evaluated at  $\underline{x} = \underline{x}^0$ ,  $\underline{x}^0$  is the set of initial parameter approximations, and  $\Delta$  is the set of corrections to the parameter approximations. Equation (22) represents a single condition equation for the model known as Adjustment by Indirect Observations. If the region of interest contains  $n$  pixels, the total set of condition equations may be written using matrix notation as

$$\underline{\ell}^0 + \underline{v} + \underline{B} \Delta = -\underline{F}(\underline{x}^0) \quad (23)$$

The stochastic information associated with the measurements is characterized by the covariance matrix  $\underline{\Sigma}$ . The corresponding cofactor matrix  $\underline{Q} = (1/\sigma_0^2) \underline{\Sigma}$  is often used for convenience, with  $\sigma_0^2$  being the reference variance. The corrections  $\Delta$  to the parameter approximations may then be calculated by solving the normal equations

$$\underline{B}^t \underline{Q}^{-1} \underline{B} \Delta = \underline{B}^t \underline{Q}^{-1} \underline{f}^0 \quad (24)$$

where  $\underline{f}^0 = -[\underline{\ell}^0 + \underline{F}(\underline{x}^0)]$ . The nonlinearity of the condition equations requires an iterative adjustment procedure, relinearizing at updated parameter values and continuing until the solution converges.

The ideal picture function,  $f(s,t)$ , can represent any picture feature that can be parametrized. For a one-dimensional edge, the parameters are the lower step edge

height  $h_1$ , the upper step edge height  $h_2$ , and the position at which this transition occurs  $x$ . For a line feature (or pulse) the parameters include  $h_1$ ,  $h_2$ ,  $x$ , and  $w$  the width of the line. In two dimensions, the parameters for an edge are  $h_1$ ,  $h_2$ ,  $x$ , and the angle,  $\theta$ , the edge makes with the  $t$ -axis. The line parameters include  $h_1$ ,  $h_2$ ,  $x$ ,  $\theta$ , and the line width,  $w$ .

Parameters of the point-spread function may also be estimated if the form of the spread is known. This research has considered two types of spread functions, rectangular and Gaussian. Given that the point-spread function is centered over each pixel both functions can be parametrized by a single parameter  $d$ .

### **2.1.3. Empirical Analysis of Moment and Least Squares Edge Locators**

Tests using both edge locating algorithms were carried out on sets of simulated one-dimensional signals containing various amounts of random noise [2]. In general, the precision of the edge estimates decreased as the noise level increased. At a noise-to-signal level of 10%, both algorithms achieved root mean square errors of less than 0.04 sampling intervals in data with an edge spread of four sampling intervals.

In addition, the two-dimensional test base used in the human observation testing were used. For the least squares implementation, errors in edge location estimates were less than 0.050 pixels for symmetric ramp-type edges with high contrast. Angle errors were typically less than one-half of a degree. Errors up to 0.20 pixels occurred in non-symmetric edge data and were consistently to the lighter side of correct edge location. These errors were a result of fitting a non-symmetric type edge function to the modeled symmetric function. Additionally, large errors were recorded from noisy data.

The grey level moment preserving method and Hueckel method [6,7,8] were applied to the same data. These methods require no iteration as in the least squares approach. However, both methods show a bias in the presence of noise which is greatest away from the center of the window of data. Therefore, both methods were implemented using an iterative procedure to ensure that the edge occurred within one pixel of the center of the window.

Errors in the edge location estimates were less than 0.050 pixels for high contrast, symmetric edge data. The Hueckel operator tends to point towards the lighter side of the edge when compared to the grey level moment preserving method. Larger errors were associated with asymmetric edges and with files which contained large amounts of noise. For noiseless cases, the two operators estimated orientation nearly perfectly with only one exception at  $22.5^\circ$  with error of  $0.8^\circ$ .

### **2.1.4. Spatial Moment Based Two-Dimensional Edge Operator**

An edge detector and locator which has been developed by Reeves, et. al. (Appendix C) matches a circular section of an image to an ideal step edge model using two-dimensional moments. This method requires no iteration and can locate edges to sub-



pixel accuracy. In addition, this method is much simpler to implement than the Hueckel operator and appears to allow more accuracy and noise immunity.

An ideal edge model is characterized by four parameters  $h$ ,  $k$ ,  $\ell$  and  $\theta$ . The edge is a straight line which separates two regions of constant grey values. The lower level has height  $h$  and the upper level is  $k$  higher than the lower level. The angle which the edge makes with the  $y$  axis is  $\theta$  and  $\ell$  is the distance from the center of the disk to the edge.

The moments of an image  $f(x,y)$  of order  $p+q$  are defined by

$$M_{pq} = \int \int x^p y^q f(x,y) dx dy \quad (25)$$

The disk is defined to have radius of one. Thus, the limits of integration are the unit circle i.e.,  $\sqrt{x^2+y^2} \leq 1$ .

A rotation of the disk by an angle  $\phi$  changes the moments as specified by

$$M'_{pq} = \sum_{r=0}^p \sum_{s=0}^q \begin{pmatrix} p \\ r \end{pmatrix} \begin{pmatrix} q \\ s \end{pmatrix} (-1)^{q-s} (\cos \phi)^{p-r+s} (\sin \phi)^{q+r-s} M_{p+q-r-s, r+s} \quad (26)$$

The angle  $\theta$  which is the angle of the edge is determined by

$$\theta = \tan^{-1} \frac{M_{01}}{M_{10}} \quad (27)$$

In order to determine the other edge model parameters, the moment set is rotated by the angle  $\theta$  until the potential edge is aligned with the  $y$  axis.

The location of the edge,  $\ell$ , may be derived from the rotated moments. The moments may be specified in terms of  $h$ ,  $k$  and  $\ell$  by integrating.

$$\begin{aligned} M'_{00} &= 2h \int_{-1}^1 \sqrt{1-x^2} dx + 2k \int_{\ell}^1 \sqrt{1-x^2} dx \\ &= h\pi + \frac{k\pi}{2} - k \sin^{-1}(\ell) - k\ell \sqrt{1-\ell^2} \end{aligned} \quad (28)$$

$$\begin{aligned} M'_{10} &= 2h \int_{-1}^1 x \sqrt{1-x^2} dx + 2k \int_{\ell}^1 x \sqrt{1-x^2} dx \\ &= \frac{2}{3} k \sqrt{(1-\ell^2)^3} \end{aligned} \quad (29)$$

$$M'_{20} = 2h \int_{-1}^1 x^2 \sqrt{1-x^2} dx + 2k \int_{\ell}^1 x^2 \sqrt{1-x^2} dx$$

$$= \frac{h\pi}{4} + \frac{k\pi}{8} + \frac{k\ell}{2} \sqrt{(1-\ell^2)^3} - \frac{k}{4} (\ell \sqrt{1-\ell^2} + \sin^{-1}(\ell)) \quad (30)$$

Equations (28), (29) and (30) may now be combined to solve for  $\ell$ , i.e.

$$M'_{20} = \frac{3}{4} \ell M'_{10} - \frac{1}{4} M'_{00} \quad (31)$$

$$\ell = \frac{4M'_{20} - M'_{00}}{3M'_{10}} \quad (32)$$

The ideal edge that is fit to the data does not allow for the quantization effects due to finite pixel size in the real data; i.e., the gray value is assumed constant over each pixel in the real data. This introduces a bias error in the calculated edge location. Although this bias appears significant, the calculated edge location versus true edge location is always a monotonic function, and thus a table look-up procedure can be used to subtract this bias effect and give perfect edge location results when no noise is present.

The edge operator was applied to noisy edge data. In the presence of noise, the operator was unbiased and performed quite well. Most edge locations (68%) were found to within 0.2 pixels of their correct location when the center of a circular window of 9 pixels was within 2.5 pixels of the edge and the signal-to-noise ratio was 20 db. Unfortunately, this edge operator was developed at a different time than the previous two operators and therefore was not evaluated against them.

## 2.2. Precision Cross Target Location

The main task of this effort was composed of two steps: the detection and approximate location of the cross target, and the precise determination of the center of the target using the least square algorithm. Detailed documentation of the first step can be found in Appendix D and Appendix E. Detailed documentation of the second step can be found in the aforesaid appendices and in reference [9].

The approach that was taken incorporated an automated procedure based upon pattern recognition and feature extraction techniques which would scan an entire image and produce the initial locations of all crosses. These initial values were then used by the least squares algorithm which modeled a cross feature to determine precise cross locations.

### **2.2.1. Pattern Recognition of Crosses**

A method has been developed to detect and recognize cross targets in digital aerial imagery. The method accomplishes these tasks by extracting two major features from the ground data. Local grey level maxima which correspond to possible cross targets serve as the detection feature. The Fourier descriptors of the contour of these targets provide recognition of the cross as well as approximate location, orientation, and cross size.

To implement local maxima detection, two processes are needed. First, to insure that true bright regions are detected and not those maxima that are attributable to system noise, atmospheric effects, etc., a circular convolutional low pass filter is applied to the data. Assuming that the size of the convolving filter window is smaller than the smallest expected cross, the grey level structure of the filter cross can be viewed as a local maximum in two dimensions. These points serve as input to the recognition phase of the algorithm.

The recognition phase accomplishes two tasks. First, the process discriminated buildings, road intersections, and other physical objects from crosses since these objects may be two-dimensional local maximums. Second, given that the object is a cross, the process determines the location, orientation, and size of the cross.

Recognition is accomplished using Fourier descriptors. to obtain the Fourier descriptors of an object, the original unfiltered image must be grey value thresholded to yield a binary image. If the threshold is chosen correctly, the object will be segmented from the background data. Typically, many thresholds are tried to successfully segment the data. A Fourier transform is applied to the contour or boundary of the segmented object to produce the object's Fourier coefficients. The coefficients (descriptors) are normalized for comparison to the coefficients of a "true" cross. If the descriptors match those of a cross within a specified accuracy, the object is classified as a cross. If the descriptors do not match, another grey level threshold is selected for segmentation until the descriptors match or until all the possible thresholds have been exhausted in which case the object is rejected as a cross. In addition, the Fourier coefficients determine location, orientation, and size information.

### **2.2.2. Location using Moments**

In general, the cross' grey level heights ( $h_1$  and  $h_2$ ) are distributed among neighboring pixels according to the location and orientation of the cross. Since the grey values of the pixels hold much of this information, a process which uses the grey levels of the cross as well as the general shape should do well in estimating location and orientation. The two-dimensional grey level moments meet this requirement.

A window of data is extracted from the original image. The location of the center of the window is determined from the Fourier descriptor location results (recognition routine). The two-dimensional moments of the window are calculated. The normalized

first order moment in x and in y, respectively, determine the center of mass of the window. Since the cross is symmetric, this is the final estimate of the cross location. After the moments are translated to this location, rotational moments are used to determine the angle of the cross. Finally, the background grey level is determined by the average grey level around the cross, not including the cross, and the grey level of the cross is estimated by the grey level at the center of the cross.

### **2.2.3. Location using Fourier Descriptors**

This method of determining location is similar to that used in the recognition phase of the algorithm. To determine accurate location and orientation of the cross, many grey level thresholds are used. Each grey level threshold produces a contour and therefore an estimate of the cross's location and orientation. Of those thresholds that produce acceptable Fourier descriptor results, only the best descriptor results are retained. Since the Fourier descriptor error measures the match to an ideal cross, the error may be used as a confidence number. The lower the error, the greater the confidence. This confidence number may then be used as a multiplicative weight with which to multiply the location result. Likewise, the final orientation angle is estimated using these weights, also.

Each location determined by the Fourier descriptors is a sub-pixel result. The process of using a weighted average of each contour center results in a surprisingly good center location with much more credence than from only one Fourier descriptor result. This technique is remarkably similar to the moment technique in one important way. The grey level moments use a continuum of grey values summed according to the moment basis functions to produce the center of mass, and thus the center of the cross. Likewise, the Fourier descriptors use a continuum of grey level thresholds to produce different centers of contour locations. These locations are weighted and summed to produce the location of the cross. However, only those Fourier descriptors results which match a cross are kept. This selective process removes those contours which are greatly affected by noise. The grey level moments, on the other hand, cannot be manipulated in this way, and must use all grey values.

### **2.2.4. Evaluation using Arizona Data Base**

A test image was obtained by generating cross targets on a digital image using the Arizona test data. This test data was derived from a digitized stereo model formed by two nearly vertical images taken in October 1966 near Guadalupe, Arizona. A cross was generated by integrating over that portion of the pixel which contained any part of the cross. This process is identical to sampling with a square aperture with area of one pixel. The cross targets were then superimposed on the digitized image.

A 512 by 512 segment of the digital image was used. Twenty-four cross targets were randomly selected and placed on the image as follows:

- \* Cross placement was done at arbitrary sub-pixel locations.
- \* Cross sizes with aspect ratios of 1 x 7, 1 x 10, and 1 x 13 were used. The aspect ratio related the width of one leg of the cross to the total length of the cross, as for example, 1 unit to 10 units.
- \* The widths ranged from 1 pixel to 1.5 pixels.
- \* The crosses were arbitrarily rotated at various orientation angles.

Additionally, zero-mean Gaussian random noise was added to the crosses with a standard deviation similar to the standard deviation of the image background around each cross. Two data sets were created; one where a standard deviation of 25% of the background noise was added to the crosses to simulate fiducial marks and one where no noise was added to simulate reseau marks.

Both the moment location and Fourier descriptor location routines were used to precisely locate the cross targets. All targets were detected with no false targets found. The average miss distance,  $\bar{\rho}$ , for all 24 targets in each image was calculated. The miss distance  $\rho$  is given by

$$\rho = \sqrt{\Delta x^2 + \Delta y^2} \quad (33)$$

where  $\Delta x$  is the error in the x position and  $\Delta y$  is the error in the y position. Thus,  $\rho$  represents the undirected distance between the actual cross position and the algorithm's determined cross position. As shown in Table 2, the *moment* location process exhibited average miss errors of 0.255 pixels and 0.241 pixels for the no noise and 25% additive noise cases, respectively. On the other hand, the Fourier descriptor method showed a significantly lower error of 0.073 pixels and 0.107 pixels for the no noise and 25% additive noise cases.

Moment and Fourier Descriptor Method Error		
Method	Noise	
	0%	25%
Moment	0.255 pixels	0.241 pixels
Fourier Descriptor	0.073 pixels	0.107 pixels

**Table 2.** Mean Location Error for Moment and Fourier Descriptor Methods.

These results were then passed to the least squares algorithm described in the next section.

### 2.2.5. Location using Least Squares

The least squares adjustment model for the fitting of a cross-type feature to image data was incorporated. The procedure followed in this routine was to construct the desired feature as a set of four rectangular components, which allowed for the convenient determination of partial derivatives by summing over all components. A total of five parameters were estimated:  $h_1$  the average background signal value,  $h_2$  the average target signal value,  $x, y$  geometric center of the target, and  $\theta$  the orientation of the target. The quantities  $W$ , the width, and  $L$ , the length, of the cross were assigned approximate values based on the average scale of the imagery and the size of the targets in the data base. The least squares approach does allow for the solution of all quantities, including those parameters defining the spread function. However, for these tests, the goal was to demonstrate the ability to extract positional information; the problem of determining precise target dimensions was not a primary goal.

The digital image files generated for the purpose of measuring the positions of crosses made use of the simulation package SIM, previously developed at Purdue University and described by Mikhail et. al. [10,11], with the following characteristics:

- \* SIM makes use of an augmented digital data base containing both elevation information and quantized density values from a digitized orthophotograph.
- \* SIM uses a bilinear interpolation in elevation and in grey shade, but both can be redefined easily.
- \* The data base used contained 1778 row by 1117 columns each, representing the Fort Sill area of Oklahoma.
- \* It is possible to superimpose artificial targets in the terrain model by assigning new grey shade values to specific data base elements.

In this way, such targets are included in the image synthesis process, and appear as other natural features in the resultant digital image file.

In one experiment, a set of nine image files were generated. The exterior orientation was varied, by assigning combinations of three different values of the primary rotation  $\omega$  and three different values of the tertiary rotation  $\kappa$ . Thus  $\kappa$  took on values  $0^\circ$ ,  $20^\circ$ , and  $45^\circ$ , and  $\omega$  was  $0^\circ$ ,  $5^\circ$ , and  $15^\circ$ . Within one file eight cross targets were imaged. In all cases, the ratio between the average pixel spacing and the data base element spacing was very roughly 1.0. Therefore the approximate dimensions of the crosses in the resultant images were 5 pixel's length by 1 pixel's width.

Two types of spread functions were used, the rectangular and Gaussian. The range in root mean square errors in  $x$  or  $y$  is from 0.033 to 0.086 pixels, with one case yielding the relatively high value of 0.394. In this particular instance, the initial approximation for  $\theta$  was  $0^\circ$ , when in fact the true value should have been close to  $\kappa$  ( $45^\circ$ ). This poor approximation appeared to have allowed convergence of the adjustment to a local minimum, and resultant residual in the final estimate was on the order

of  $45^\circ$  in orientation and 1.0 pixel in position. The approximation was calculated by a very simple procedure employing template matching. The use of pattern recognition and feature extraction algorithms for deriving approximations totally alleviates this problem.

The least squares algorithm was next used on the Arizona test data with the initial approximations supplied by the moment location method and then repeated with the Fourier descriptors method. Using approximations from both processes, the rms location errors were reduced in x to 0.05 pixels and in y to 0.03 pixels. Even when the moment process resulted in a poorer location than the Fourier descriptor process, the least squares routine still converged to virtually the same result. However, the least squares routine required at least one more iteration per data point when it started with the moment process approximations.

Many features in aerial images are smaller than the imaging aperture used to digitize the scene. Yet, these digitized features can still be located by observers. Therefore, the intent of the next experiment was to determine if objects with subpixel features could be accurately located.

A test image was constructed with twenty-five thin crosses superimposed on the Arizona data base. In order to simulate thin features, the cross leg widths ranged from 0.5 to 1.0 pixels and the aspect ratios were 1x7, 1x10, and 1x13. The Fourier descriptor algorithm was used to detect these crosses in the image. Of the original 25 crosses, only 17 crosses were detected. Due to the thinner features, recognition was not established by the local maximum routine or the Fourier descriptor routine for eight crosses. However, no false detections were made. Increasing the number of detected crosses by lowering the local maximum or Fourier descriptor thresholds will also increase the number of false targets.

Of those crosses detected, the Fourier descriptor method resulted in root mean square errors in X of 0.335 pixels and in Y of 0.317 pixels (Table 3). Using the Fourier descriptor results as initial approximations, the least squares algorithm improved the location results to 0.048 and 0.036 root mean square error (pixels) in X and Y, respectively, as shown in Table 3.

Location of Thin Crosses		
Method	X rms (pixels)	Y rms (pixels)
Fourier Descriptors	0.335	0.317
Least Squares	0.048	0.036

**Table 3.** Location results on crosses with subpixel features.

### 3. Effects of Image Processing on Geometric Fidelity

Provided that the Fourier descriptor-least squares algorithm exhibited minimal location errors, digital image processing techniques were monitored for their effect on geometric fidelity. The types of processing work that were studied included: image compression, image enhancement, and image resampling using various interpolation functions.

#### 3.1. Effects of Digital Image Compression -- Cosine Compression

For each of the two Arizona test images, a two-dimensional adaptive cosine transform compression scheme was applied. The image was sub-divided into 16 by 16 sub-blocks and the coefficients resulting from the 2-D Direct Cosine Transform were quantized using a bit assignment scheme based on the energy in each block [12]. The quantization levels were determined from the desired compression. The resulting images were then reconstructed using 8, 2, 1, and 0.5 bits/pixel. For each of the eight possible image files described above, two different processing procedures were used:

- \* An algorithm based on Fourier descriptors and moments was used for detection and location, followed by the least squares algorithm for precise positioning.
- \* An algorithm based only on the Fourier descriptors was used followed by the least squares algorithm.

In general, as the number of bits decreased the location of the crosses changed implying geometric shift. Furthermore, some of the crosses lost an entire leg due to the sub-division performed by the compression. In this event, the algorithm did not recognize these crosses and therefore the total number of crosses was reduced. Processing of the original images showed that the location of a cross could be achieved with a precision of 0.03 - 0.05 pixels. Compression to 2 bits/pixel led to 0.06 - 0.13 pixels; 1 bit/pixel led to 0.16 - 0.18 pixels; and 0.5 bits/pixel led to 0.36 - 0.71 pixels. These results are summarized in Table 4 for easy reference.

Precision after Cosine Compression	
Compression (bits/pixel)	Precision (pixels)
8 (no processing)	0.03 - 0.05
2	0.06 - 0.13
1	0.16 - 0.18
0.5	0.36 - 0.71

**Table 4.** Precision results after cosine compression at various compression levels.



### 3.2. Effects of Mean and Median Filters in the Presence of Noise

In the previous section, the test data consisted of artificial cross data embedded on real terrain data. The noise statistics of the terrain data due to the system and atmospheric effects were unknown. To better isolate the effects of processing, new test data was artificially generated with known noise statistics. The base image consisted of 49 crosses with the aspect ratio 1 x 7 oriented at random angles and placed at random sub-pixel locations on a flat field. The width of the legs of each cross was set at 3 pixels, thereby making each cross 21 pixels in length. This larger size was necessary to prevent the median filter from removing large portions of the crosses. Three additional images were created by adding varying amounts of independent zero-mean Gaussian random noise to this base image. The standard deviation of the noise was set at 20%, 40%, and 60% of the center step height (grey value) of the cross.

On each of the above images, eight separate processes were performed. These processes included a 3 by 3 mean filter, a 3 by 3 median filter, a circular mean filter with diameter of 3, and a circular median filter with a diameter of 3. The above processes were repeated using 5 by 5 window and diameters of 5 pixels. To the authors' knowledge, the circular median filter has yet to be introduced in the literature. For this reason, a brief discussion of the circular median filter is given in Appendix F.

To the four test images, the Fourier descriptor precision method was applied. This method was preferred over the moment method due to its superior performance on the Arizona test set. Due to the large cross size, the least squares algorithm was not applied to the test data. Notably, for the zero noise case, the average error is 0.061 pixels. In general, due to the off-pixel boundary placement of the crosses and the arbitrary angle of each, the Fourier descriptor location result was not identically zero.

**Median Results** For the 20% and 40% noise cases, the median filters improved the location result only slightly in comparison with the non-filtered result (Table 5). Median filtering in the 60% case resulted in a marked reduction in the error by almost 40%. All the cases have shown virtually no difference in the location result between the square and circular median filters. Thus, the differences between the two types of median filters at size 3 on the geometric accuracy was minimal at most. For the size 5 case, filtering caused deterioration, as there was a slight increase in location error over that of the non-filtered data. The window size was at times too large for the feature. As the noise level was increased, the resulting location error was increased. Only in the highest noise case did the median filter increase the accuracy. Additionally, the circular median resulted in less error than the square median. This can be attributed to the additional bias the square median filter had on orientation.

Mean Location Error (pixels) for Synthetic Test Image									
Noise	Before Processing	Median				Mean			
		Square		Circular		Square		Circular	
		3x3	5x5	d=3	d=5	3x3	5x5	d=3	d=5
0%	0.061	0.078	0.091	0.078	0.081	0.026	0.043	0.026	0.033
20%	0.108	0.093	0.114	0.094	0.105	0.087	0.118	0.081	0.124
40%	0.207	0.188	0.255	0.184	0.239	0.184	0.258	0.185	0.250
60%	0.404	0.249	0.359	0.242	0.304	0.220	0.300	0.214	0.279

**Table 5** Mean location error in pixels for synthetic test image using Fourier descriptor method. Square and circular median and mean filter results are shown using window sizes of 3 and 5 pixels in the presence of various amounts of noise.

In general, given that the feature was much larger than the window size of the filter, the median improved the accuracy of the Fourier descriptor location result on noisy data. However, when the feature was not larger than the window size, the median filter appeared to distort the feature and the accuracy of the Fourier descriptor location result was reduced.

**Mean Results** For all cases, the location results of the Fourier descriptors were better after using the mean filter than prior to this preprocessing (Table 5). Notably, for the zero noise case, the location was improved greatly. This was not seen in the median case where in only the noisier cases did the location result improve over that of the unprocessed result. Additionally, all cases of the mean filter of size 3, the location result was better than the result for the median of the same size. Also, there was no discernible difference in location accuracy between the square and circular means. Therefore, smoothing the data improved the accuracy of the Fourier descriptors and should be considered as a preprocessing operation for the Fourier descriptor method.

### 3.3. Effects of Resampling Using Various Interpolation Functions

The effects caused by resampling on metric accuracy was studied. A linear resampling scheme was used with no scaling of the coordinate axes. In a continuous domain, all distance measures are preserved after this type of transformation. This transformation can be preformed by rotating and translating the original image. In practice, the inverse transformation is performed on the new image position to determine the corresponding old image position. The calculated old image position in general is not located on a pixel boundary. Various interpolation functions have been proposed to extract the correct grey level value from this non-integer position. Three common interpolation functions were considered: biconstant (nearest neighbor), bilinear, and bicubic. The biconstant function merely uses the closest pixel's grey value. The bilinear function interpolates from the four nearest pixels' grey values. The bicubic

function incorporates the sixteen nearest pixels' grey values.

The Arizona test data was resampled at various rotations, and horizontal and vertical translations. The translation intervals were 0.125, 0.25, and 0.375 pixels and every combination of these intervals in X and Y were performed to yield 9 different translated images. The rotation intervals were 12.5°, 25°, and 37.5° and each rotation interval was applied to the nine translated images. This yielded 27 different resampled images. The transformed images contained 19 to 22 crosses depending on the amount of rotation. Therefore, the number of crosses that were resampled was at least 513. This process was performed for each interpolation function. The precision location algorithm using Fourier descriptors was applied to each image. The mean error horizontally and vertically was tabulated as well as the mean radial error or miss distance  $\rho$  for each interpolation function. Table 6 summarizes the results.

Locations Results (pixels) with various Interpolation Functions					
Interpolation	Mean	12.5°	25.0°	37.5°	All
Biconstant	x	-0.0317	-0.0359	-0.0479	-0.0383
	y	0.0067	0.0014	0.0289	0.0121
	$\rho$	0.1629	0.1795	0.1952	0.1787
	# crosses	202	188	186	576
Bilinear	x	0.0018	-0.0108	-0.0349	-0.0141
	y	0.0363	0.0457	0.0469	0.0427
	$\rho$	0.1289	0.1461	0.1434	0.1392
	# crosses	207	189	189	585
Bicubic	x	0.0017	-0.0136	-0.0430	-0.0177
	y	0.0552	0.0762	0.0738	0.0680
	$\rho$	0.1638	0.1823	0.1858	0.1769
	# crosses	207	189	189	585

**Table 6.** Location results (pixels) after using the three interpolation functions: biconstant, bilinear, and bicubic. There are nine different translations for every rotation angle given in the table.

The mean errors in the horizontal and vertical directions for all interpolations were virtually zero and therefore show no significant bias. The values appear to show the X error to favor the negative side of the correct position and the Y error to favor the positive side of the correct position. However, since the Fourier descriptor method does not give perfect location results in noiseless data, it is felt that this small bias could stem from the location method rather than factors associated with the interpolation functions.

For all cases  $(X, Y, \rho)$ , as the rotation angle was increased the location error increased. Since the digitization and interpolation schemes are based on a square grid structure, it can be inferred that the location error increases as the rotation approaches  $45^\circ$  and then decreases as the angle approaches  $90^\circ$ . This observation suggests that if the interpolation functions were implemented using a circularly symmetric window, location error would not be influenced by the rotation angle.

Lastly, the radial mean errors for biconstant and bicubic interpolations were very nearly the same at all rotations. In addition, the radial mean accuracy was significantly better using the bilinear interpolation function and was consistently better at all orientations. Since bicubic interpolation uses more information than bilinear interpolation (16 grey levels as compared to 4 grey levels), it was assumed that bicubic interpolation would produce less metric distortion. This now appears not to be the case. From these results, position information can best be obtained only from the four nearest pixel locations. Thus, bilinear interpolation produces less metric distortion than either biconstant or bicubic interpolation. However, more work is needed to understand the factors underlying this result.

## REFERENCES

1. Unruh, J.E. and Mikhail, E.M., "Digital Image Simulation for Photogrammetric Applications", Technical Report 1981, DMA/AC, St. Louis, MO. Contract No. DMA 700-75-C-0119.
2. Thurgood, J.D. and Mikhail, E.M., "Subpixel Mensuration of Photogrammetric Targets in Digital Images", CE-PH-82-2, School of Civil Engineering, Purdue University, August 1982.
3. Tabatabai, A.Y., "Edge Location and Data Compression for Digital Imagery", Ph.D. thesis, School of Electrical Engineering, Purdue University, December 1981.
4. Thurgood, J.D., "Subpixel Mensuration of Photogrammetric Targets in Digital Images", Ph.D. thesis, School of Civil Engineering, Purdue University, December 1982.
5. Thurgood, J.D. and Mikhail, E.M., "Photogrammetric Aspects of Digital Images", presented at ASP meeting, Denver, CO. March 14-20, 1982.
6. Hueckel, M.F., "An Operator which Locates Edges in Digitized Pictures", J. Ass. Comput. Mach., vol. 18, pp. 113-125, 1971.
7. Hueckel, M.F., "A Local Operator which Recognizes Edges and Lines", J. Ass. Comput. Mach., vol. 20, pp. 634-647, 1973.
8. Hueckel, M.F., "Erratum for [4]", J. Ass. Comput. Mach., vol. 21, p. 350, Apr. 1974.
9. Mikhail, E.M., "Photogrammetric Target Location to Subpixel Accuracy in Digital Images", presented at the 39th Photogrammetric Week, Stuttgart, West Germany, Sept. 19-24, 1983.
10. Mikhail, E.M. Unruh, J.E., and Alspaugh, D.H., "Image simulation from Digital Data", 1977 Proceedings of the Fall Technical Meeting of the American Society of Photogrammetry.
11. Unruh, J.E., Alspaugh, D.H. and Mikhail, E.M., "Sensor Simulation from Spectral and Digital Terrain Data", Final Technical Report for the Defense Mapping Agency, Aerospace Center, St. Louis, MO. DMA Contract 700-75-C-0119, 1977.
12. Chen, W.H. and Smith, C.H., "Adaptive Coding of Monochrome and Color Images", IEEE Transactions on Communications, Vol. COM-25, Nov. 1977, pp. 1285-1292.

APPENDIX A

PHOTOGRAMMETRIC ANALYSIS OF DIGITAL IMAGES

J. D. Thurgood and E. M. Mikhail

Proceedings of Symposium of Commission III  
The International Society of Photogrammetry and Remote Sensing

Helsinki, Finland

June, 1982

# PHOTOGRAMMETRIC ANALYSIS OF DIGITAL IMAGES

J. D. Thurgood and E. M. Mikhail

Purdue University, West Lafayette, Indiana 47907, U.S.A.

## ABSTRACT

Recent experiments using hardcopy measurement of digital images have shown that accuracies significantly less than the pixel size are attainable when pointing to well-defined features such as dots, crosses and edges. In addition, digital processing techniques for the automatic extraction of feature location to subpixel levels have been applied to simulated aerial images with significant results. This paper describes approaches to the photogrammetric analysis of digital images which investigate the role of the human observer and digital processing in the extraction of precise geometric information.

## 1. INTRODUCTION

The growth in the use of digital images has been accompanied by the development of digital processing techniques in many areas: from image enhancement and image coding, to pattern recognition and image classification procedures. From the photogrammetrist's point of view, the primary concern must be the geometric fidelity of the image, that is, the utility of the stored data for the extraction of geometric information. This information takes the form of image coordinates, measured lengths and areas, or corresponding quantities in a three-dimensional space formed using overlapping imageries with different perspectives. However, the bulk of recent work in digital image processing has dealt with gray shade(density or intensity) values, but not particularly with the effect of changes in gray shade distribution on the ability to extract precise metric information. As digital images become increasingly available in the form of either directly-acquired records or as digitized photographs, it is important that measurement processes be developed commensurate with the full potential of the image data.

Makarovic and Tempfli (4) considered the photogrammetric problem in terms of both pictorial and metric requirements, using the sampling theorem as a basis for the former, and using the sampling interval as a basis for the latter.

When considering specific image features such as edges and lines, there do exist well-known digital processing techniques for their detection and location to pixel levels

(7). Other research (1,2) has developed theoretical bounds on the variance of positional estimates using digital images, without actually implementing the necessary algorithms to solve the estimation problem. The method devised by Hueckel (3) has been until recently the only means of obtaining edge and line location estimates to subpixel levels, without applying interpolation.

Recent work by Unruh and Mikhail (10) involved the measurement of digital images written on film with pixel sizes of 25, 50, and 100  $\mu\text{m}$ . The digital image files were synthesized aerial images, produced using the program SIM. This program makes use of a digital terrain model containing gray shade information to generate images exhibiting all the perspective characteristics of an aerial image, but in a digital form (6). SIM allowed the introduction of artificial image targets, either by modification of the gray shades in the terrain model, or by superimposing targets in the image itself. Results from these and similar experiments carried out with digitized aerial photographs, indicated that hardcopy digital images could provide measurement precisions of 7  $\mu\text{m}$  or better for monoscopic viewing and roughly 10  $\mu\text{m}$  for stereoscopic viewing of targets appearing in both images, even with pixel sizes of up to 50  $\mu\text{m}$ .

This paper will first review a new method of locating an edge to subpixel levels using moment preserving, developed at Purdue University by Tabatabai and Mitchell. Secondly, an approach to the image modelling problem which uses the method of least squares adjustment to estimate feature position is described. Then summaries of two experiments being conducted at Purdue University into the geometric analysis of digital images will be given: one involving the hardcopy measurement of images containing edges, and another implementing the least squares algorithm to automatically locate cross targets in simulated aerial imageries.

## 2. EDGE LOCATION USING MOMENT PRESERVING

The method of edge location by moment preserving is described in more detail in (8).

For simplicity, let us first consider the one-dimensional case, in which an attempt is being made to model a set of data to an ideal step edge as shown in Figure 1. The three parameters defining the edge are:  $h_1$  the signal value below the edge,  $h_2$  the signal value above the edge, and  $X$  the location of the edge. Moment preserving is used as the criterion of best fit of a set  $\underline{d}$  of  $n$  data points to the ideal edge  $f(s)$ . Rather than solve directly for  $X$ , the edge location is defined as  $k + \frac{1}{2}$ , where  $k$  is the (unknown) number of samples below the edge. Since there are three unknowns,



we set the first three sample moments equal to those associated with the ideal edge, that is:

$$\bar{m}_j = \frac{k}{n} h_1^j + \frac{n-k}{n} h_2^j \quad \text{for } j = 1, 2, 3 \dots \dots \dots (1)$$

where

$$\bar{m}_j = \frac{1}{n} \sum_{i=1}^n I_i^j \dots \dots \dots (2)$$

is the  $j$ -th sample moment, and  $j$  is a power. The three equations given by (1) may be solved directly in a closed form. In particular, the solution for  $k$  is given by

$$k = (n/2) \{1 - c / \sqrt{4 + c^2}\} \dots \dots \dots (3)$$

where

$$c = \{3\bar{m}_1\bar{m}_2 - \bar{m}_3 - 2(\bar{m}_1)^3\} / \sigma^3$$

is the skewness of the data, and  $\sigma^2 = \bar{m}_2 - \bar{m}_1^2$ . From equation (3) it is clear that  $k$  need not be an integer, and therefore sub-pixel edge location is obtained directly. This method of edge location assumes that the data consists of monotonically increasing values. This will not be the case if noise is present, and preprocessing of the data to smooth out oscillations due to noise has been shown to improve results by a significant amount. Extension of the model to two dimensions necessitates, in addition to moment preserving, calculations to determine the center of gravity of the image data, in order to solve for the two parameters which now define a straight edge passing through the image. Moment preserving is very simple to apply, and yields unbiased estimates if the edge lies near the center of the area considered. Biased results may be overcome by recentering the area to be modelled after an initial solution.

### 3. LEAST SQUARES ADJUSTMENT IN IMAGE MODELLING

Let  $f(s,t)$  represent the output of a perfect imaging system, that is, the ideal picture function. Consider next a linear, spatially-invariant imaging system with a normalized point-spread function  $p(s,t)$  assumed known. Then let  $I(s,t)$  denote a random variable representing the measurement at sampling position  $(s,t)$ . We may model the measured quantity using the convolution

$$I(s,t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi,\eta) p(s-\xi, t-\eta) d\xi d\eta \dots \dots \dots (4)$$

Consider now a set of  $u$  parameters  $\underline{x}$  which completely characterizes  $f(s,t)$  over the region of interest. Equation (4) may be rewritten as

$$\mathbb{I}(s,t) - f(s,t;\underline{x}) * p(s,t) = 0 \dots\dots\dots(5)$$

where  $*$  denotes the convolution operation. Then for the  $ij$ -th picture element which is a sample of  $\mathbb{I}(s,t)$  at  $s=s_i$ ,  $t=t_j$ , we may write a linearized condition equation of the form (dropping  $s,t$  for simplicity)

$$\mathbb{I}_{ij}^0 + v_{ij} + \underline{B}_{ij} \underline{\Delta} = - F_{ij}(\underline{x}^0) \dots\dots\dots(6)$$

where  $\mathbb{I}_{ij}^0$  is the initial estimate for the observation,  
 $v_{ij}$  is the measurement residual,  
 $F_{ij}(\underline{x}) = - f_{ij}(\underline{x}) * p_{ij}$ ,  
 $\underline{B}_{ij}$  is the set of partial derivatives of  $F_{ij}(\underline{x})$  with respect to the parameters, evaluated at  $\underline{x}=\underline{x}^0$ ,  
 $\underline{x}^0$  is the set of initial parameter approximations, and  
 $\underline{\Delta}$  is the set of corrections to the parameter approximations.

Equation (6) represents a single condition equation for the model known as Adjustment by Indirect Observations. The total set of equations can then be solved by forming the normal equations in the conventional manner (5).

Tests using both edge-locating algorithms were carried out on sets of simulated one-dimensional signals containing random noise at various levels by these authors (9). In general, the precision of the edge estimates decreased as the noise level increased. At a noise-to-signal level of 10%, both algorithms achieved root mean square errors in  $\lambda$  of less than 0.40 sampling intervals, in data with a spread of four sampling intervals across the edge.

#### 4. HARDCOPY MEASUREMENT OF IMAGES CONTAINING EDGES

##### 4.1. Data generation

In order to allow the fullest control over the geometric and densitometric nature of the images to be measured, simulated data was generated which represented a set of fifteen digital image files. Each file consisted of a two-dimensional array of size 1024 by 1024 pixels, containing a gray shade distribution modelling a main image area and a set of four cross fiducial or reference marks. The functions defining the edges were selected in order to allow the effects of various edge characteristics to be studied: Width (0,2,4 pixels), Type (step,ramp,exponential,raised cosine), Contrast (high,low), Noise (0,20%,40%), and Orientation (0,20°,45°).

The fifteen files were written on film as a single image, shown in Figure 2, using an Optronics precision film-writer located at DMA/AC St.Louis with a  $25\mu\text{m}$  square aperture. We shall refer to each image file as an individual frame. The data was written so that the gray shade values corresponded to a roughly linear scale in density in the final film positive.

#### 4.2. Measurement of the hardcopy image

Measurements to the reference marks and to the edges were performed by three observers. A total of three replications were made of the whole experiment. The measurement instrument used was the Zeiss PSK Stereo Comparator, operated in the single-plate binocular viewing mode. Within one frame a total of ten pointings were made: one to each of the reference marks and three each to two distinct locations on the edge itself. The total set of data acquired was comprised of 1350 pairs of coordinates.

#### 4.3. Preliminary processing and analysis of measurements

The preliminary analysis of all data gathered included the estimation of precision associated with the ability of each observer to measure the fiducial marks and the transformation of the comparator measurements into an image reference coordinate system.

Using repeat measurements, an estimate of the repeatability of each observer in pointing to the fiducial marks was determined, over eighty measurements, as  $1.1\mu\text{m}$ ,  $3.2\mu\text{m}$  and  $2.3\mu\text{m}$  for observers 1,2 and 3 respectively. These expressed the 68% confidence limit of a measurement in x or in y.

The measured fiducials were fitted to the control points for each frame individually, in order to localize any systematic deformation in the film. Statistical tests involving the relative fit of the two-dimensional conformal (four-parameter) and the affine (six-parameter) transformations to the control points, indicated that in general the four-parameter transformation modelled the measured points adequately. A lack of fit detected in certain frames was attributed to residual film deformations of the order of  $3\text{--}4\mu\text{m}$  in magnitude. It was decided that insufficient information existed to attempt to further model these residuals.

#### 4.4. Analysis of edge measurements

A detailed statistical analysis was performed on the errors in measurement to the edge within each frame. In order to provide a common reference, the error in a single

measurement was calculated as the normal distance from the measured point to an idealized edge location which was defined using the mean value of gray shade between the light and dark sides of the frame. These error values were negative if they lay to the lighter side of the ideal edge, and positive if they lay to the darker side.

An analysis of variance (ANOVA) carried out with the factors Replicate, Observer, Frame, and Location, indicated that all four significantly affected the accuracy of the edge measurements. In addition, many of the second- and higher-order interaction terms appeared as significant. This meant in effect that the variability in measurements within a single group of three pointings to the same location was very small when compared with all other groups of measurements. The variability between replicates and between locations is due in part to the fact that the observer was not consistent in his edge-pointing process over time, and in part to the irrecoverable errors introduced by film deformations. Then for a practical situation, where a given edge location is measured only once, we may obtain a better idea of the variability in edge measurement by pooling the variability associated with Replicate and Location. The mean errors in measurement to the edge within each frame are shown in Figure 3. These means were calculated over all 18 measurements to the same edge by individual observers. Also shown are the 68% confidence levels associated with a single measurement.

As can be seen, not only was the difference in measured edge position large between observers, but there was also a consistent trend: observer 1 measured to the lighter side of observer 3, who measured to the lighter side of observer 2. This was an indication of how the judgement of an individual in locating the edge played an important role, independently of the characteristics of the edge.

Considering specific frames, it is obvious that the accuracy in pointing varied considerably. It was possible to perform comparisons between certain frames and groups of frames, in order to examine the effects of a number of edge characteristics. For example, it was found that the mean position in pointing to edges with a spread width of two pixels (frames 2,3,6) was roughly 0.3 pixels ( $7.5 \mu\text{m}$ ) different from the mean position in pointing to edges with four pixels' width (frames 4,5,7). Significant differences in precision were noted when comparing frames 1 (high contrast step edge) and 10 (low contrast step edge). In this case, the mean measured position remained almost identical, but the standard deviation of a single measurement rose from an average of 0.12 pixels ( $3.0 \mu\text{m}$ ), to 0.35 pixels ( $8.8 \mu\text{m}$ ) for frame 10. However, in the case of added random noise (frames 13,14,15), both precision and

accuracy were affected. Edge orientation and edge type (ramps versus others such as exponential and raised cosine) were also found to be significant for at least one of the three observers. Overall, the precision in edge measurements ranged from 0.08 pixels (2.0  $\mu\text{m}$ ) for observer 1, frame 2, to 0.47 pixels (11.8  $\mu\text{m}$ ) for observer 2, frame 10.

Analysis of these results is continuing, but it is obvious from these preliminary findings that both accuracy and precision in measurement may be affected by various edge characteristics.

## 5. MEASUREMENT OF CROSS TARGETS IN SIMULATED AERIAL IMAGERY

The digital image files generated for the purpose of measuring the positions of crosses made use of the simulation package SIM previously mentioned, and a subroutine implementing the least squares modelling algorithm.

### 5.1. Implementation of the cross pointing algorithm

The least squares adjustment model for the fitting of a cross-type feature to image data was incorporated in a FORTRAN subroutine named POINT4. The procedure followed in this routine was to construct the desired feature as a set of (up to four) rectangular components, which allowed for the convenient determination of partial derivatives by summing over all components. A total of five parameters were estimates in POINT4 (see Figure 4):

- $h_1$  the average background signal value,
- $h_2$  the average target signal value,
- $X, Y$  defining the geometric center of the target, and
- $\theta$  the orientation of the target with respect to the rows and columns of the image.

POINT4 allowed the use of a rectangular or Gaussian spread function. For the functional model to be correct, it was required that the function be circularly-symmetric and separable into two equivalent one-dimensional functions. However, if the spread function were not circularly-symmetric, the errors introduced would not be expected to be significant for the case of a symmetric target such as the cross. For these tests, the width of the spread function was assigned so that at least 90% of the spread lay within half a sampling interval of the pixel center.

Quantities  $W$ , the width, and  $L$ , the length, of the cross, were assigned approximate values based on the average scale of the imagery and the size of the targets in the data base.

The least squares approach does allow for the solution of all quantities, including those parameters defining the spread function. However, for these preliminary tests, the goal was to demonstrate the ability to extract positional information. The problem of determining precise target dimensions was not the primary one in this context.

## 5.2. Tests with synthetic vertical frame images

For these tests, four image files were generated using SIM, corresponding to two vertical stereo pairs, A and B, with 60% overlap. Each file contained sixteen segments of fifty rows by fifty columns, positioned so that the image of a cross target lay near their center. The cross targets were formed by modifying the data base elements as shown in Figure 5(a). The two stereopairs differed basically only in scale, so that on pair A the crosses were imaged with size roughly six pixels across, and on pair B the crosses appeared three pixels across. These files were identical to those used in the hardcopy analyses of Unruh and Mikhail.

Parameter approximations for each target were obtained through the use of a simple matching template and a cross correlation procedure. The template used represented a cross of length five pixels as shown in Figure 6. Then, considering the image data to be denoted by a function  $g$ , and the template function (taking values either zero or one) to be  $t$ , the cross correlation function  $C_{tg}$  was calculated at each template position  $(m,n)$ , using

$$C_{tg} = \sum_{i=-2}^{+2} \sum_{j=-2}^{+2} t_{ij} g_{i+m, j+n}$$

The template position yielding a maximum value of  $C_{tg}$  was used as the initial approximation for the position of the cross. This then defined an eight row by eight column neighborhood of image data which was involved in the adjustment proper.

Approximations for the background and target signal values were calculated by examining the average gray shade 1) at the corners, and 2) in the center, of this array.

The quantity  $\theta$  was carried as an unknown in the adjustment, but the corresponding approximation  $\theta^0$  was always assigned the value zero. In the event that target dimensions were also to be determined, similar methods involving the use of cross-correlation functions could be implemented to calculate initial parameter approximations. In these tests, values  $W=2, L=6$  were used for pair A, and  $W=1, L=3$  for pair B. For each target the adjusted position within the eight by eight pixel area was referenced back to the overall image

coordinate system. The true error was then estimated as the computed position minus the ideal position. Root mean square errors were formed over the sixteen targets. In addition, the root mean square error of the orientation estimates was determined. These results are shown in Table 1, along with the corresponding measures of accuracy from the hardcopy measurement of the same targets (on image files A-2 and B-2).

The estimated standard deviation in estimates  $\hat{x}$  or  $\hat{y}$  range from .029 to .065 pixel. Using these values as a measure of performance, the adjustment model with a rectangular spread function did slightly better than the model with a Gaussian spread function.

Estimates of position containing errors larger than  $3 \times \text{RMSE}$  were found to be associated with target 11 in image A-1, target 2 in image A-2 and target 10 in image B-1. It is thought that these relatively large errors (of magnitude up to .200 pixel) may have been caused by noise in the background image being confounded with the cross image. With these points removed from the calculation of the summary statistics, the RMSE's varied from .020 to .033 for the rectangular spread model, and from .026 to .052 for the Gaussian spread model.

In all four image files, the accuracies obtained were substantially higher than for the corresponding hardcopy human measurements using film pixel sizes of 25 and 50  $\mu\text{m}$ .

All adjustments but one converged satisfactorily, generally within four iterations. In one case, the adjusted parameter values oscillated back and forth with successive relinearizations, causing large changes in the reference variance and the orientation estimate, but changes in estimated target position of less than .010 pixel. This instability is still being investigated, and is thought to be associated with the near-singularity of the normal coefficient matrix  $\underline{N}$ .

### 5.3. Tests with synthetic images of variable tilt and orientation

Next, a set of nine image files were generated, with an effective camera station equal to that for image B-1 as used previously. In this experiment however, the exterior orientation was varied, by assigning combinations of three different values of the primary rotation  $\omega$  and of the tertiary rotation  $\kappa$ :  $\kappa$  took on values 0, 20° and 45°, and  $\omega$  0, 5° and 15°. Within one file, sixteen targets were imaged. In all cases the ratio (average pixel spacing)/(data base element spacing) was very roughly 1.0. The two distinct target types were as follows: targets 1 to

8 were 3 pixels across (type 1), and targets 9 to 16 were 5 pixels across (type 2), formed as shown in Figures 5 a) and b) respectively. Initial parameter approximations were made in a similar way to before, but for  $\theta^0$  using a set of six templates for matching, each one corresponding to feature rotations roughly  $15^\circ$ - $20^\circ$  apart. Image target dimensions were assigned as  $W=1$  for all cases, and  $L=3$  for type 1,  $L=5$  for type 2 targets.

The adjustment results are shown in Table 2.

As previously, the model with the Gaussian spread did not perform quite as well as the model with a rectangular spread. The most significant variations in algorithm performance appear to be a function of the exterior orientation and of the target type.

Firstly, the accuracy of the target position estimates was highest for the case where  $\omega$  equals zero. This was to be expected, since only the image rotation rather than image displacement due to tilt, could be modelled within the adjustment. Then the quantity  $\theta$  corresponded to an estimate of  $\kappa$ , whereas no estimate could be made for additional tilts. In general, the accuracy in pointing to the 3 by 3 pixel crosses decreased as  $\kappa$  increased from zero: the RMSE in  $\bar{X}$  or  $\bar{Y}$  increased from .021 pixel for  $\kappa=0$ ,  $\omega=0$ , to .158 pixel for  $\kappa=45^\circ$ ,  $\omega=5^\circ$ . This increase was accompanied by an increase in the errors in orientation estimates.

Secondly, in pointing to target type 2, the RMSE values actually decreased as  $\kappa$  increased from zero, and the accuracy of the orientation estimates remains roughly constant. This is still being examined.

Both types of adjustment, that is, pointing to both types of target, appeared to be adversely affected when poor approximations were made for target orientation. Since the larger target allowed a more accurate determination of the approximation  $\theta^0$ , in most cases the final estimate for  $\theta$  was close to the  $\kappa$  value. The exception was for the case where  $\kappa=45^\circ$  and  $\omega=15^\circ$ , where the final estimate for the rotation of target 11 was close to zero. The resultant error in position was on the order of one pixel. All values in parentheses in Table 2 are the corresponding statistics for the case where the known  $\kappa$  values were used as approximations for  $\theta$ . As can be seen, the accuracy in pointing to target type 1 is increased substantially for the images with large  $\kappa$  values. For target type 2, no improvement was made over the original adjustment cases, with the exception of the case described earlier.



## 6. CONCLUSIONS

The results described above indicate that both human and algorithmic operators are capable of extracting high-precision geometric information from digital images, provided that the targets to be measured are sufficiently well-defined.

In measurements of hardcopy digital image files containing edges, the precision in pointing to these straight edges was found to vary from 2.0  $\mu\text{m}$  to 11.8  $\mu\text{m}$  (68% confidence interval). Variation in precision was associated with observer and with edge characteristics such as contrast, additive image noise and the width of the edge-spread. Accuracy was more difficult to interpret, since the problem associated with determining exactly what the human measures is still unsolved. However, consistent biases in pointing across the profile of the edge were associated with each observer. The analysis of how these biases relate to an 'ideal' edge location are continuing.

The analytical approach was found to work well in the case of cross-pointing in simulated aerial imagery. Crosses of size three pixels across were located with accuracies of .03-.06 pixel in vertical imagery. In imagery with a variety of orientations, the approximation for the cross orientation was critical in guaranteeing the highest accuracies. Thus for the three pixel crosses, the accuracies ranged from .02 pixel for vertical image files to .16 pixel for files containing crosses with a 45° orientation. A second type of cross, of size five pixels across, was large enough to allow a reliable approximation to be determined for the orientation, thus allowing an accuracy in the range .03-.06 pixel to be maintained, in general, even with up to 15° of tilt present.

Factors such as the size of the target and the background contrast play important roles in both types of operator. Further effort is required in both areas, that is, to gain a better understanding of how the human observer performs the pointing process, and to use all information possible to train an automatic digital operator to perform the same tasks to the highest degrees of precision and accuracy.

## ACKNOWLEDGEMENTS

The work described in this paper has been supported by the U.S. Army Research Office under Contract No DAAG 29-81-K-0063. The authors express their gratitude to Dr. Steven Mock, the Scientific Program Director. The authors would also like to thank Professor O. R. Mitchell of the School of Electrical Engineering at Purdue University for his assistance, and Dr. J. E. Unruh of the Defense Mapping

Agency Aerospace Center for the use of the Optronics Film-writer and for his useful suggestions.

#### REFERENCES

1. Gonsalves, R.A., 1976, Cramer-Rao bounds on mensuration errors, Applied Optics, Vol. 15, No. 5, pp.1270-1275.
2. Helstrom, C.W., 1970, Resolvability of Objects from the Standpoint of Statistical Parameter Estimation, J. Optical Society of America, Vol. 60, No. 5, pp.659-666.
3. Hueckel, M.H., 1971, An Operator which Locates Edges in Digitized Pictures, J. Ass. Comput. Mach., Vol. 18, pp.113-125.
4. Makarovic, B. and Tempfli, K., 1979, Digitizing Images for Automatic Processing in Photogrammetry, ITC Journal, 1979-1, pp.107-126.
5. Mikhail, E.M., 1976, Observations and Least Squares, IEP New York.
6. Mikhail, E.M., Unruh, J.E., and Alspaugh, D.H., 1977, Image Simulation from Digital Data, 1977 Proceedings of the Fall Technical Meeting of the American Society of Photogrammetry.
7. Rosenfeld, A., and Kak, A.C., 1976, Digital Picture Processing, Academic Press, New York.
8. Tabatabai, A.J. and Mitchell, O.R., 1981, Edge Location to Subpixel Accuracy in Digital Imagery, submitted to IEEE Trans. on Pattern Analysis and Machine Intelligence.
9. Thurgood, J.D. and Mikhail, E.M., 1982, Photogrammetric Aspects of Digital Images, Proceedings of the 48th Annual Meeting of the American Society of Photogrammetry, pp.295-304.
10. Unruh, J.E. and Mikhail, E.M., 1981, Digital Image Simulation for Photogrammetric Applications, Technical Report, School of Civil Engineering, Purdue University, CE-PH-81-1.

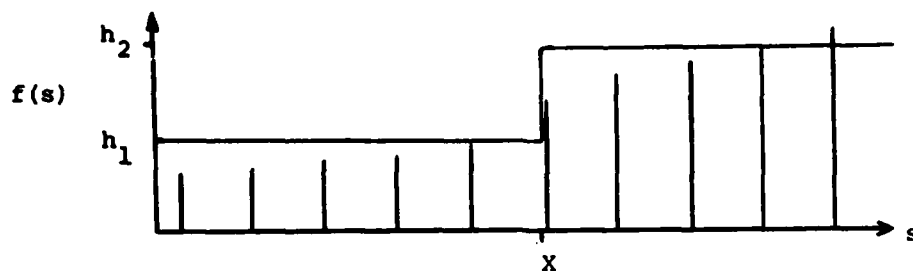


Figure 1. Ideal edge function.

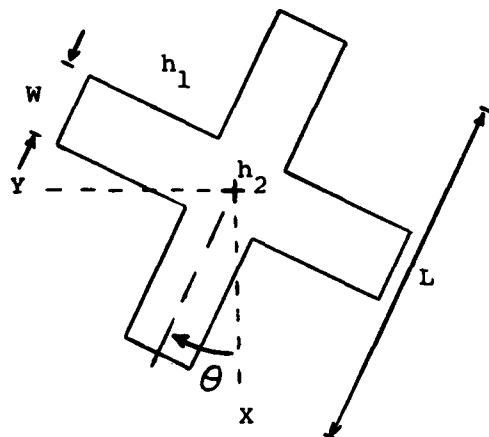
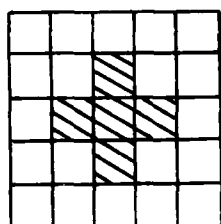
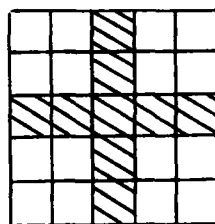


Figure 4. Cross-type feature.



a) Type 1



b) Type 2

Figure 5. Data base modification for cross targets.

0	0	1	0	0
0	0	1	0	0
1	1	1	1	1
0	0	1	0	0
0	0	1	0	0

Figure 6. Matching template for  
non-rotated target.

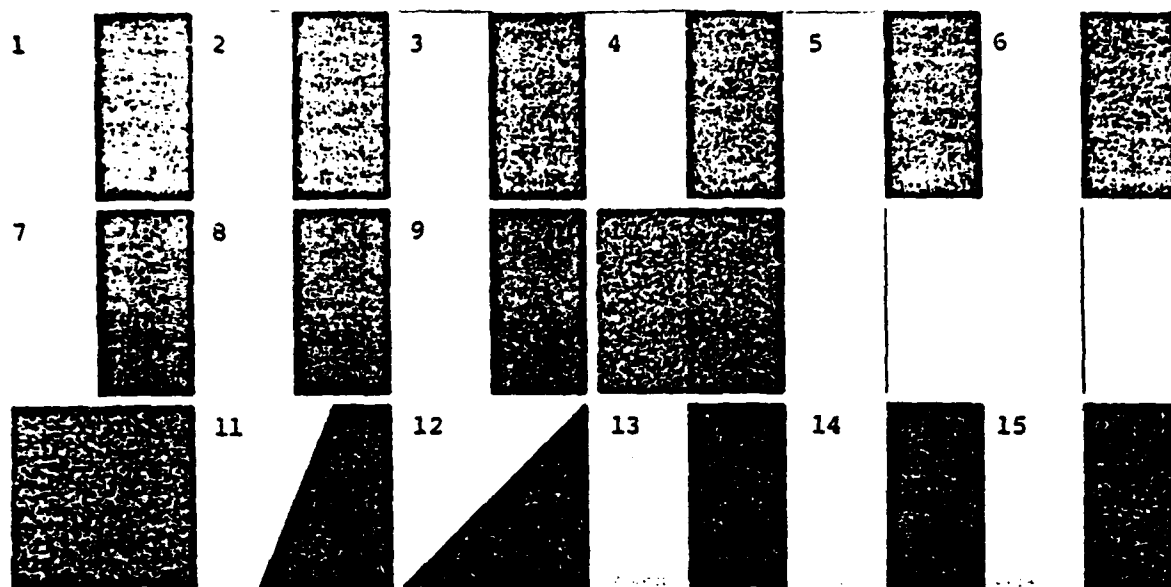


Figure 2. Image file containing edges.

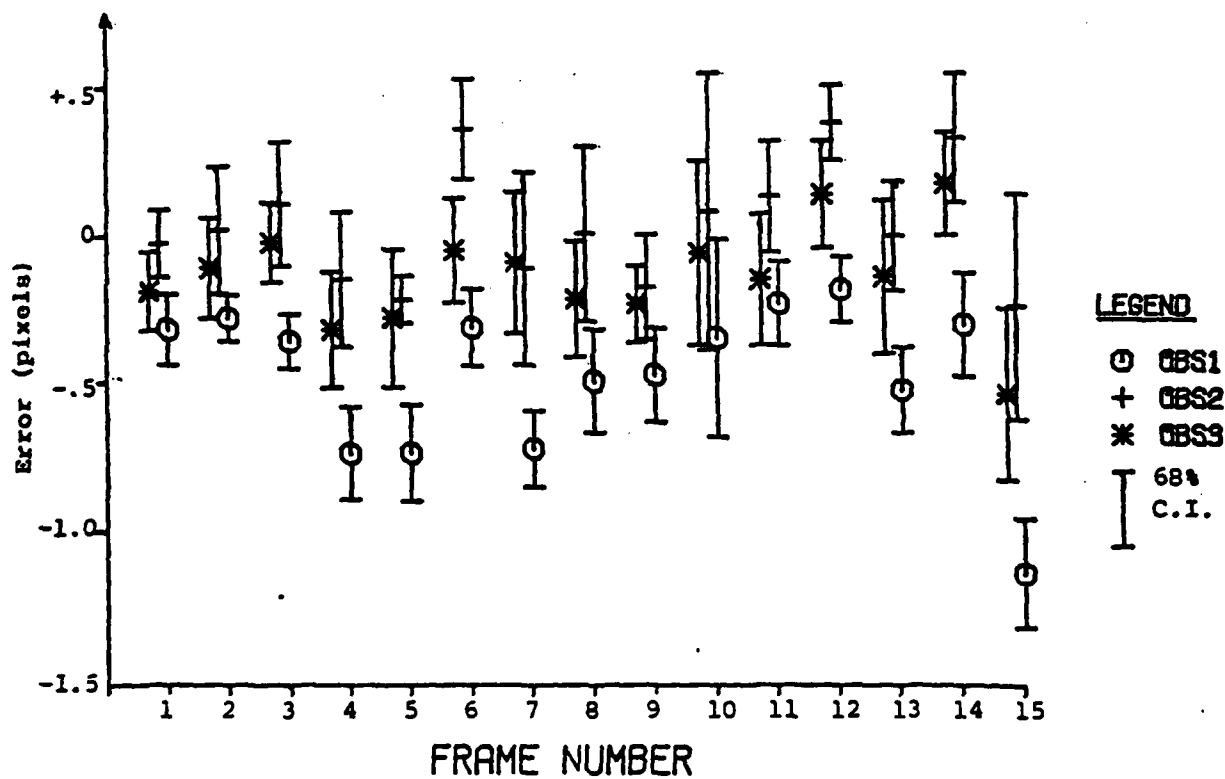


Figure 3. Hardcopy edge measurements.

Table 1 Cross pointing on vertical imagery.

IMAGE FILE	RECT. SPREAD MODEL		GAUS. SPREAD MODEL		LARGE ERROR	HARDCOPY RMSE( $\hat{X}$ or $\hat{Y}$ ) (pixels)
	RMSE( $\hat{X}$ or $\hat{Y}$ ) (pixels)	RMSE( $\hat{\theta}$ ) (degrees)	RMSE( $\hat{X}$ or $\hat{Y}$ ) (pixels)	RMSE( $\hat{\theta}$ ) (degrees)		
A-1	.030(.020)	1.54(0.39)	.033(.028)	0.46(0.33)	#11	-
A-2	.063(.023)	2.76(0.75)	.065(.026)	2.67(0.63)	# 2	.292
B-1	.029(.021)	1.21(1.21)	.051(.043)	1.31(1.33)	#10	-
B-2	.033	1.46	.052	1.86	-	.290

Figures in parentheses represent values calculated after removing points containing large errors (positional error of more than 3\*RMSE).

Table 2 Cross pointing on imagery with various orientations.

		RECT. SPREAD MODEL		GAUS. SPREAD MODEL	
IMAGE $\kappa$ $\omega$		RMSE( $\hat{X}$ or $\hat{Y}$ ) (pixels)	RMSE( $\hat{\theta}$ ) (degrees)	RMSE( $\hat{X}$ or $\hat{Y}$ ) (pixels)	RMSE( $\hat{\theta}$ ) (degrees)
<u>Type 1 target</u>					
0	0°	.021	1.67	.041	1.33
	5°	.032	2.74	.046	3.02
	15°	.053	3.27	.068	2.94
20°	0°	.081(.048)	8.61(4.23)	.063(.052)	4.19(2.28)
	5°	.070(.079)	7.82(8.56)	.084(.095)	7.45(3.53)
	15°	.085(.085)	8.99(8.74)	.102(.113)	10.72(8.66)
45°	0°	.090(.026)	22.19(2.15)	.097(.028)	22.58(4.19)
	5°	.158(.049)	43.72(1.87)	.171(.050)	43.80(1.33)
	15°	.086(.090)	32.55(5.11)	.134(.112)	35.83(4.16)
<u>Type 2 target</u>					
0	0°	.030	1.50	.052	1.05
	5°	.108	1.19	.114	1.79
	15°	.105	3.24	.132	2.80
20°	0°	.021(.022)	0.79(0.57)	.026(.026)	0.71(0.71)
	5°	.048(.049)	1.65(1.65)	.048(.049)	1.47(1.48)
	15°	.049(.051)	4.57(4.57)	.059(.059)	4.64(4.91)
45°	0°	.028(.028)	1.15(1.13)	.029(.029)	1.24(1.24)
	5°	.028(.028)	1.67(1.59)	.021(.021)	1.55(1.55)
	15°	.277(.031)	14.69(2.59)	.267(.019)	14.97(2.84)

Figures in parentheses represent values obtained when assigning correct approximations for  $\theta$ .

APPENDIX B

EDGE LOCATION TO SUBPIXEL ACCURACY IN DIGITAL IMAGERY

A. Y. Tabatabai and O. R. Mitchell

IEEE Transactions on Pattern Analysis and Machine Intelligence

Vol. PAMI-6, No. 2

March, 1984

# Edge Location to Subpixel Values in Digital Imagery

ALI J. TABATABAI AND O. ROBERT MITCHELL, SENIOR MEMBER, IEEE

**Abstract**—A new method for locating edges in digital data to subpixel values and which is invariant to additive and multiplicative changes in the data is presented. For one-dimensional edge patterns an ideal edge is fit to the data by matching moments. It is shown that the edge location is related to the so-called "Christoffel numbers." Also presented is the study of the effect of additive noise on edge location. The method is extended to include two-dimensional edge patterns where a line equation is derived to locate an edge. This in turn is compared with the standard Hueckel edge operator. An application of the new edge operator as an edge detector is also provided and is compared with Sobel and Hueckel edge detectors in presence and absence of noise.

**Index Terms**—Edge detection, edge location, Hueckel operator, moments.

## I. INTRODUCTION

RECENTLY there has been a growing trend toward collecting and processing terrain images in digital form. While there has been considerable work done on general digital image processing in such areas as image coding, image restoration, and feature extraction, there has been little or no attention paid to the effects of such processing on the geometric fidelity of the image. The problem is motivated by the need for accurate measurements from remotely sensed imagery, which is of prime importance to the mapping communities [1], [2]. Many of these images are in digital form. Thus, photogrammetric analysis which deals mostly with metric aspects of images must be combined with digital image processing and feature extraction procedures, such as edge detection and location techniques.

In this paper, an attempt has been made to give an objective analytical definition to the term "edge location," when a blurred and noisy observation of an ideal edge in a digitized picture is given. It shall be assumed throughout this work that the desired image has been sampled and quantized to obtain an acceptable discrete representation of the continuous image.

There appears to be a lack of a quantitative and universally accepted definition for the term "edge location." The most applicable publications which have dealt with the above problem are those by Hueckel [3]–[5]. Hueckel assumes two-dimensional data are available and fits a parametric function to the empirically obtained edge disk so that the Euclidean

distance is minimized. The parameters obtained can be used to estimate the edge location to subpixel accuracy. No analytical study of the performance of this operator, in the presence of noise has been presented.

Frei and Chen [6] have developed an algorithm where an "ideal edge element" is defined as a straight boundary line passing through the center of a  $3 \times 3$  window, thus separating two regions of different, constant intensities  $h_1, h_2$ . They have characterized the "ideal edge element" by its magnitude,  $|h_1 - h_2|$  and orientation. However, for the purpose of "edge location," the "ideal edge element" can also be easily characterized by the equation of the boundary line. One obvious disadvantage of this approach is the constraint imposed on the line by forcing it to pass through the center of the window. Another possible disadvantage of the algorithm is its sensitivity to noise.

The Frei and Chen algorithm is a generalization of the so called "gradient" edge detection technique, where different edge detection methods correspond to different numerical approximations to the gradient.

Roberts [7] defines the gradient  $G$ , at point  $(i, j)$  as

$$G = |f(i, j) - f(i + 1, j + 1)| + |f(i + 1, j) - f(i, j + 1)| \quad (1)$$

or equivalently,

$$G = |\langle F, W_1 \rangle| + |\langle F, W_2 \rangle| \quad (2)$$

where

$$F = \begin{bmatrix} f(i, j) & f(i, j + 1) \\ f(i + 1, j) & f(i + 1, j + 1) \end{bmatrix}$$

$$W_1 = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$$

$$W_2 = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$$

and

$\langle A, B \rangle \triangleq$  inner product of matrices  $A$  and  $B$

$$\langle A, B \rangle = \sum_{i=1}^N \sum_{j=1}^N a_{i,j} b_{i,j}$$

Sobel [8] uses (1) to approximate the gradient  $G$ ; but instead of a  $2 \times 2$  intensity matrix  $F$ , a  $3 \times 3$  intensity matrix is used, and in his case the weighting matrices  $W_1, W_2$  are de-

Manuscript received January 25, 1982; revised May 31, 1983. This work was supported by the U.S. Army Research Office.

A. J. Tabatabai was with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907. He is now with Bell Laboratories, Holmdel, NJ 07733.

O. R. Mitchell is with the School of Electrical Engineering, Purdue University, West Lafayette, IN 47907.

defined as

$$W_1 = \begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \quad W_2 = \begin{bmatrix} 1 & 0 & -1 \\ 2 & 0 & -2 \\ 1 & 0 & -1 \end{bmatrix} \quad (3)$$

One should note that the "gradient" method cannot be used if subpixel accuracy of edge location is desired, unless some kind of interpolation process is performed on the data matrix.

Machuca and Gilbert [9] have given a theoretical argument, based upon noise models that "gradient" methods should not be used. Instead they have proposed a new algorithm where the edge detection is not based on derivatives but uses integrals to reduce the effect of noise. Their approach uses moments to detect edges [10]. They have also shown that preprocessing of data can appreciably improve the performance of their operator.

Burnett and Huang [11] have proposed a simple statistical description of signals with step edges. Such signals are graphically represented by a path through a trellis. Blurred versions of these signals are similarly represented. Then a cost or length is assigned to each branch of the trellis and a MAP sequence estimate of the signal is computed by finding the minimum cost or length path through the trellis. The Viterbi algorithm is introduced as an efficient means to find the minimum cost or length path through the trellis. The estimates produced by this algorithm are then used for edge location measurement with "pixel" accuracy.

Finally, Jacobus and Chien [12] have presented two edge detection techniques which are based on the application of arrays of edge detectors, each sensitive to a different group of edge types. These techniques are claimed to be able to measure an edge with subpixel accuracy.

A good review of some of the techniques mentioned above is presented in [13]-[16].

#### Summary of Problems Considered

As a first step in studying the metric properties of digital pictures a new one-dimensional analytical formulation of edge location is presented in Section II. The approach is based on fitting an ideal edge to a set of empirically obtained one-dimensional edge data, such that the first three sample moments are preserved. It will be shown that the parameters of the ideal edge are related to the Gauss-Jacobi mechanical quadrature problem, where the edge location is related to the so-called "Christoffel numbers," and intensity levels are the zeros of the orthogonal polynomial associated with input probability density function. Also the effect of additive uncorrelated, Gaussian noise on edge location is studied. Finally, the effects of averaging and median filtering the input edge data are studied, and the reduction of noise effects due to preprocessing is empirically verified. Section III describes the extension of the edge location method to two-dimensional edge data where the problem becomes more complicated by the fact that at least two parameters should be specified to describe edge location. The new operator is compared with the standard Hueckel operator. Also included is an application of the new operator as an edge detector and a comparison with Hueckel and Sobel edge detectors.

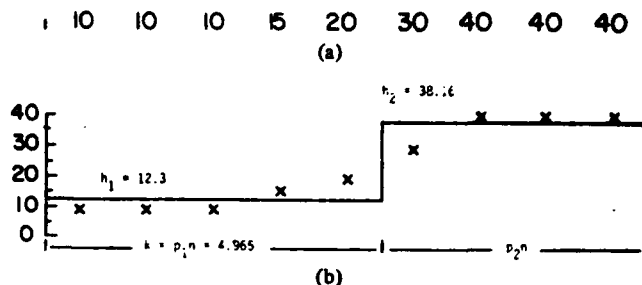


Fig. 1. Empirical edge pattern as input to the operator, and the ideal edge as its output. (a) Input empirical data. (b) Sample data and ideal edge that matches first three sample moments.

## II. A NEW ONE-DIMENSIONAL EDGE OPERATOR

Recent trends in mapping communities toward digitization of analog pictures have motivated the need for accurate measurements from digital images. In an ideal case, reflected light intensities from an object and its background will be piecewise constant with discontinuities at the edges. Hence edges and their location play a central role on the study of metric fidelity of digital images. When the edge data is digitized, it may be possible to determine the edge location to subpixel accuracy by examining the transfer function of the digitizing equipment and the output pixel values.

As was described earlier, there are a number of techniques available that can be used, if pixel accuracy of measurement is desired (see [6]-[11]). But for a more accurate measurement, Hueckel [3]-[5] has provided the only alternative. However, the Hueckel operator requires two-dimensional data.

In this section, an analytical definition of edge location is given. It is shown that the method is easy to derive in closed form. This in turn significantly reduces computational load. This method is quite insensitive to the sequence length. The method is invariant to multiplicative and additive changes in the data. This is important because many optical, photographic, and digital image processing operations can scale and shift the data. Throughout this section it is implicitly assumed that empirical edge data is the output of a sampled scan line across an edge pattern and consists of a discrete, one-dimensional sequence of numbers.

A sampled scan line across a step edge in the absence of noise is characterized by a set of numbers  $x_i$ 's,  $i = 1, 2, \dots, n$ , that are either monotonically nondecreasing or nonincreasing. On the other hand, an ideal edge is a sequence of one brightness value  $h_1$ , followed by a sequence of another brightness value  $h_2$ . Here we define an operator that, when applied to empirically obtained edge data [shown in Fig. 1(a)], generates an ideal edge [shown in Fig. 1(b)], such that the first three sample moments of the input data sequence which are defined as

$$\bar{m}_i = \frac{1}{n} \sum_{j=1}^n x_j^i \quad i = 1, 2, 3 \quad (4)$$

are preserved. If we let  $k$  denote the number of  $h_1$  values in the ideal edge, then preserving the first three sample moments between the input and output sequences is equivalent to solv-



ing the three equations

$$\sum_{j=1}^2 p_j h_j^k = \bar{m}_k \quad (5)$$

where

$$p_1 = \frac{k}{n}$$

and

$$\sum_{j=1}^2 p_j = 1$$

for three unknowns  $p_1$ ,  $h_1$ , and  $h_2$ .

The solutions to these equations are presented in [17] and are given by

$$h_1 = \bar{m}_1 - \bar{\sigma} \sqrt{\frac{p_2}{p_1}} \quad (6)$$

$$h_2 = \bar{m}_1 + \bar{\sigma} \sqrt{\frac{p_1}{p_2}} \quad (7)$$

$$p_2 = \frac{1}{2} \left[ 1 + \bar{s} \sqrt{\frac{1}{4 + \bar{s}^2}} \right] \quad (8)$$

where

$$\bar{s} = \frac{m_3 + 2\bar{m}_1^3 - 3\bar{m}_1 m_2}{\bar{\sigma}^3} \quad (9)$$

$$\bar{\sigma}^2 = \bar{m}_2 - \bar{m}_1^2. \quad (10)$$

From the above results we see that  $k = np_1$  may be a non-integer [see Fig. 1(b)]; hence, if we define the edge location "γ" as

$$\gamma = k \quad (11)$$

where the first pixel is located at  $j = \frac{1}{2}$  and subsequent pixels have spacing of one, then we are able to obtain a subpixel measurement of edge location (e.g., edge location need not be at a sample point).

In general subpixel accuracy of measurements is not possible without first introducing some sort of interpolation process. This, however, is not necessary if the method mentioned above is used. Furthermore the computation involved in this process is much less than the classical two-dimensional operator of Hueckel [3]-[5].

Table I shows the applicability of this technique to different edge patterns. Rows (b) and (c) in Table I provide an example which shows the insensitivity of this method to input sequence length.

It is easily shown that the quantity  $\bar{s}$  defined in (9) is equal to sample skewness (see [21]) of the input data sequence. Thus,

$$\bar{s} = \frac{1}{n} \sum_{i=1}^n \frac{(x_i - \bar{m}_1)^3}{\bar{\sigma}^3} \quad (12)$$

Therefore, one can conclude that the edge location is a function of skewness only, a useful result that will have several applications.

TABLE I  
EDGE LOCATION FOR DIFFERENT INPUT EDGE PATTERNS. THE FIRST NUMBER IN EACH SEQUENCE REPRESENTS THE VALUE AT LOCATION  $j$ . SUBSEQUENT NUMBERS HAVE A SPACING OF ONE.

Input sequence $\{x_i\}$	Calculated Edge Location
(a) 0 0 0 5 1 1 1 1 1	3.508
(b) 0 0 0 .25 1 1 1 1 1	3.800
(c) 0 0 0 .25 1 1 1 1 1 1 1 1 1 1 1	3.861
(d) 0 1 2 3 4 6 8 1 1	4.987

Furthermore, the edge location given in (11) is invariant under scaling and translation (i.e., multiplicative and additive changes in sample values). Thus if

$$z_i = ax_i + b \quad i = 1, 2, \dots, n \quad (13)$$

where  $a$  and  $b$  are any constants, the sample skewness of the  $z_i$  sequence is identical to that of the  $x_i$  sequence, and the measured edge location (11) will be identical for both. This can be significant because many photographic and digital processing stages may scale the data in this manner.

### III. EXTENSION TO MULTIPLE EDGE DATA

In the previous section the fitting of an ideal edge to a sequence of empirical edge data was discussed. In this section we will show the extension of the technique to multiple edge patterns.

Assume the sequence  $x_i$ ,  $i = 1, 2, \dots, n$ , represents a multiple edge pattern. If it is desired to fit  $N \ll n$  ideal brightness levels ( $N - 1$  ideal edges) to the sequence  $\{x_i\}$ , then the edge operator should preserve the first " $2N - 1$ " sample moments.

Hence, the following " $2N$ " equations should be satisfied:

$$\sum_{j=1}^N p_j h_j^k = \bar{m}_k \quad k = 0, 1, 2, \dots, 2N - 1 \quad (14)$$

where

$h_j \triangleq j$ th brightness level associated with  $N$  level ideal edge.

$p_j \triangleq$  relative frequency of  $h_j$  among the  $N$  brightness levels.

$\bar{m}_0 \triangleq 1$ .

Szego [18] has provided a solution to the above equations under the context of Gauss-Jacobi mechanical quadrature problem, where  $h_j$ ,  $j = 1, 2, \dots, N$ , are the roots of the polynomial

$$g(h) = \begin{vmatrix} \bar{m}_0 & \bar{m}_1 & \bar{m}_2 & \cdots & \bar{m}_N \\ \bar{m}_1 & \bar{m}_2 & \bar{m}_3 & \cdots & \bar{m}_{N+1} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ \bar{m}_{N-1} & \bar{m}_N & \bar{m}_{N+1} & \cdots & \bar{m}_{2N-1} \\ 1 & h & h^2 & \cdots & h^N \end{vmatrix} \quad (15)$$

$$= \Delta_N h^N + \Delta_{N-1} h^{N-1} + \cdots + \Delta_0 \quad (16)$$

where (16) is an expansion of (15) by minors

$$m_0 \triangleq 1 \quad (17)$$

and  $p_j$ 's,  $j = 1, 2, \dots, n$ , are the so-called "Christoffel numbers," obtained by

$$p_j^{-1} = \sum_{m=0}^N \psi_m^2(h_j) \quad j = 1, 2, \dots, N \quad (18)$$

where  $\{\psi_m(x) \in \pi_N\}$ ,  $m = 0, 1, \dots, N$ , are orthonormal polynomials associated with the input data distribution

$$f(x) = \frac{1}{N} \sum_{k=1}^N \delta(x - x_k). \quad (19)$$

It can be shown [19] that the solution presented above is identical to solving the system of equations:

$$\begin{aligned} C_0 \bar{m}_0 + C_1 \bar{m}_1 + \dots + C_{N-1} \bar{m}_{N-1} &= -\bar{m}_{2N-1} \\ C_0 \bar{m}_1 + C_1 \bar{m}_2 + \dots + C_{N-1} \bar{m}_N &= -\bar{m}_{N+1} \\ &\vdots \\ C_0 \bar{m}_{N-1} + C_1 \bar{m}_N + \dots + C_{N-1} \bar{m}_{2N-2} &= -\bar{m}_{2N-1}. \end{aligned} \quad (20)$$

Szego [18] has shown that the system matrix

$$\begin{bmatrix} \bar{m}_0 & \bar{m}_1 & \bar{m}_2 & \dots & \bar{m}_{N-1} \\ \bar{m}_1 & \bar{m}_2 & & \dots & \bar{m}_N \\ & & \ddots & & \\ \bar{m}_{N-1} & \bar{m}_N & & \dots & \bar{m}_{2N-2} \end{bmatrix} \quad (21)$$

is positive definite if there are at least  $N$  distinct values among  $\{x_i\}_{i=1}^N$ , and this in turn implies that equations (20) have a solution. Once the values of  $C_0, C_1, \dots, C_N$  are found, the step levels  $h_1, h_2, \dots, h_N$  can be found from the roots of polynomial

$$w(x) = x^N + C_{N-1} x^{N-1} + \dots + C_1 x + C_0. \quad (22)$$

It can be shown [19] that the roots of (22) are identical to the roots of (16) and that the roots are all real valued and are different from each other if there are at least  $N$  distinct values among the  $x_i$ .

Once we have found the  $N$  values of  $h_j$  the  $N$  values of  $p_j$ , can be found from the  $N$  equations

$$\sum_{j=1}^N p_j h_j^k = \bar{m}_k \quad k = 0, 1, 2, \dots, N-1. \quad (23)$$

The theoretical steps, and the results derived in this section were not based on the actual empirical edge data shape. The method will obviously work if one is trying to fit an ideal step edge to a set of nondecreasing empirical edge data shown in Fig. 2(a). But if the empirical edge data pattern is as shown in Fig. 2(b), the data must be broken into two monotonic pieces and each edge found separately.

#### IV. RELATIONSHIP BETWEEN ESTIMATING AN IDEAL EDGE AND ESTIMATING A PROBABILITY DISTRIBUTION FUNCTION

So far, our objective has been to fit an ideal edge to a set of empirical edge data, under the assumption that they form a monotonically nondecreasing sequence. In this case an ideal edge was characterized by two adjacent step levels  $h_1$  and  $h_2$  and their relative frequency of occurrence  $p_1$  and  $p_2$  where  $p_1 + p_2 = 1$ . But the ideal edge can also be regarded as repre-

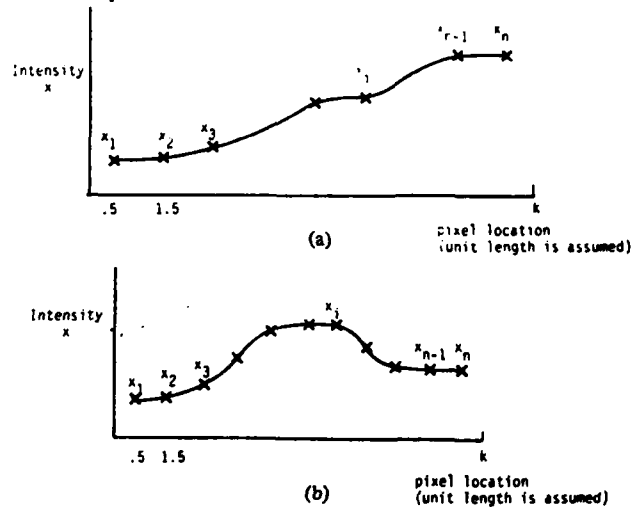


Fig. 2. Examples of multiple edge patterns.

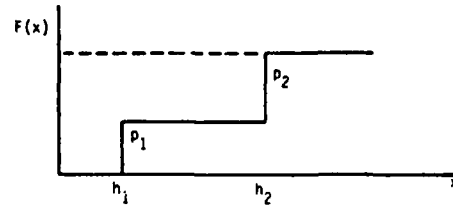


Fig. 3. Relation of ideal edge parameters and a probability distribution function.

senting a distribution function where  $h_1$  and  $h_2$  are the two abscissa values (in general  $N$  abscissa values for  $N$ -level multiple ideal edge) where jumps of heights  $p_1$  and  $p_2$ , respectively, occur ( $N-1$  jumps for the  $N$ -level multiple edge). See Fig. 3 for an example distribution function.

Estimation of a distribution function has been a well-defined problem in statistics, and many techniques have been presented over the years to solve the so-called reduced or finite problem of moments, i.e., the problem of determining or approximating a probability distribution from a finite number of its moments. Hill [20] has described the more common of the existing methods and has presented an explicit procedure for utilizing them numerically.

Von Mises step function approximation [21] is similar to the technique we use to fit an ideal  $N$ -dimensional edge to a set of empirical input edge data by preserving the first " $2N-1$ " moments. If one knew the original edge data was not an ideal edge, it might be possible to fit a curve other than the ideal step edge to the data based on the sample moments. Hill [19] and Ederton and Johnson [22] give example parametric curves used for fitting to various shapes of distribution functions.

#### V. STUDY ON EFFECT OF ADDITIVE NOISE ON EDGE LOCATION

In this section we shall discuss the effect of additive, zero-mean, Gaussian noise on edge location. The analytical model is assumed to be

$$Z_i = x_i + W_i \quad i = 1, 2, \dots, n \quad (24)$$

where

$x_i$  =  $i$ th sample of the original input data

$W_i$  = independent identically distributed zero-mean, Gaussian random variable.

with

$$E\{W_i W_j\} = \sigma_w^2 \quad i = j \\ = 0 \quad i \neq j$$

and

$Z_i$  =  $i$ th observed value.

An exact solution to the edge location jitter due to noise defined as

$$\epsilon = n(\hat{p} - p) \quad (25)$$

where

$\hat{p}$  = measured relative frequency

$p$  = true relative frequency

can be obtained in the following way. From (8) and (9) we have

$$\hat{p} - p = \left( S \sqrt{\frac{1}{4 + S^2}} - \bar{s} \sqrt{\frac{1}{4 + \bar{s}^2}} \right) / 2 \quad (26)$$

where

$$S = \frac{M_3 + 2M_1^3 - 3M_1 M_2}{\hat{\sigma}^3} \quad (27)$$

$$M_i = \frac{1}{n} \sum_{k=1}^n Z_k^i = \frac{1}{n} \sum_{k=1}^n (x_k + W_k)^i \quad i = 1, 2, 3 \quad (28)$$

$$\hat{\sigma}^2 = M_2 - M_1^2 \quad (29)$$

and  $\bar{s}$  is the sample skewness of the noiseless edge as given in (9). As can be seen from (27)–(29),  $S$  is a function of  $W_1, W_2, W_3, \dots, W_n$ ; therefore the mean-square edge jitter is

$$E\{(\hat{p} - p)^2\} = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} (\hat{p} - p)^2 f_{\vec{W}} < (w_1, w_2, \dots, w_N) dw_1 dw_2 \dots dw_n \quad (30)$$

where  $f_{\vec{W}}(w_1, w_2, \dots, w_n)$  is the joint probability density function of i.i.d. Gaussian random variables. Hence

$$f_{\vec{W}}(w_1, w_2, \dots, w_n) = \frac{1}{(2\pi\sigma_w^2)^{n/2}} e^{-(\vec{W}^T \vec{W})/2\sigma_w^2} \quad (31)$$

where

$$\vec{W} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_n \end{bmatrix}$$

Substitution of (31) in (30) yields

$$E\{(\hat{p} - p)^2\} = \frac{1}{(2\pi\sigma_w^2)^{n/2}} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \dots \int_{-\infty}^{\infty} \left( S \sqrt{\frac{1}{4 + S^2}} - p \right)^2 e^{-\vec{W}^T \vec{W}/2\sigma_w^2} dw_1 dw_2 \dots dw_n \quad (32)$$

Evaluation of (32) is a Herculean task, therefore our effort is mainly focused toward finding some analytical solution for edge location movement when some approximations are made in the evaluation of sample skewness.

In general, one can write

$$M_i = E\{M_i\} + Y_i \quad i = 1, 2, 3 \quad (33)$$

where

$$E\{Y_i\} = 0$$

and

$$\text{var}\{Y_i\} = \text{var}\{M_i\}.$$

Writing the above explicitly for random variable  $M_1$ , will yield

$$Y_1 = \frac{1}{n} \sum_{i=1}^n W_i \quad (34)$$

$$E\{M_1\} = \bar{m}_1 \quad (35)$$

$$\text{var}\{M_1\} = \frac{\sigma_w^2}{n} \quad (36)$$

For random variables  $M_2$ , and  $M_3$  we have

$$Y_2 = \frac{1}{n} \sum_{i=1}^n (W_i^2 + 2x_i W_i - \sigma_w^2) \quad (37)$$

$$Y_3 = \frac{1}{n} \sum_{i=1}^n (W_i^3 + 3x_i^2 W_i + 3x_i W_i^2 - 3x_i \sigma_w^2) \quad (38)$$

$$E\{M_2\} = E\left\{\frac{1}{n} \sum_{i=1}^n (x_i + W_i)^2\right\} = \bar{m}_2 + \sigma_w^2 \quad (39)$$

$$E\{M_3\} = \bar{m}_3 + 3\bar{m}_1 \sigma_w^2 \quad (40)$$

Calculation of the variances of  $M_2$  and  $M_3$  is more cumbersome [19] but the following results can be obtained:

$$\text{var}\{M_2\} = (2\bar{m}_2 + \sigma_w^2) \frac{2\sigma_w^2}{n} \quad (41)$$

$$\text{var}\{M_3\} = (3\bar{m}_4 + 12\bar{m}_2 \sigma_w^2 + 5\sigma_w^4) \frac{3\sigma_w^2}{n} \quad (42)$$

In order to calculate edge location, one needs to calculate the "skewness" of the observed data. From (9)

$$S = \frac{M_3 + 2M_1^3 - 3M_1 M_2}{\hat{\sigma}^3} \quad (43)$$

where

$$\hat{\sigma}^2 = W_2 - W_1^2.$$

Combining (33) and (43),

$$S = \frac{\bar{m}_3 + 3\bar{m}_1 \sigma_w^2 + Y_3 + 2(\bar{m}_1 + Y_1)^3 - 3(\bar{m}_1 + Y_1)(\bar{m}_2 + \sigma_w^2 + Y_2)}{(\bar{\sigma}^2 + \sigma_w^2 + Y_2 - Y_1^2 - 2\bar{m}_1 Y_1)^{3/2}}. \quad (44)$$

According to (13), with no loss of generality, one may assume

$$\bar{m}_1 = 0 \quad (45)$$

$$\bar{m}_2 = 1. \quad (46)$$

Then the observed skewness can be approximated as

$$S = \sigma^3 \bar{s} + \sigma^3 Y_3 - 3\sigma Y_1 \quad (47)$$

where

$$\sigma = \frac{1}{(1 + \sigma_w^2)^{1/2}} \quad (48)$$

and the terms  $Y_1^3$ ,  $Y_1 Y_2$ ,  $Y_2^2$  are disregarded for the sake of simplicity and due to small effect they have on  $S$ . This later statement is verified when experimental and theoretical results are compared. Substitution of (47) and (28) into (27) results in

$$\epsilon = \frac{n}{2} \left[ \left( \bar{s} \sqrt{\frac{1}{4 + \bar{s}^2}} - \sigma^3 \bar{s} \sqrt{\frac{1}{4 + \sigma^6 \bar{s}^2}} - \sigma^3 Y_3 \sqrt{\frac{1}{4 + \sigma^6 \bar{s}^2}} + 3\sigma Y_1 \sqrt{\frac{1}{4 + \sigma^6 \bar{s}^2}} \right) \right]. \quad (49)$$

The mean-square error associated with edge location can then be calculated as

$$E\{\epsilon^2\} = \frac{n^2}{4} \left[ \left( \sqrt{\frac{1}{4 + \bar{s}^2}} - \sigma^3 \sqrt{\frac{1}{4 + \sigma^6 \bar{s}^2}} \right)^2 \bar{s}^2 + \frac{\sigma^2}{4 + \sigma^6 \bar{s}^2} (\sigma^4 E\{Y_3^2\} + 9E\{Y_1^2\} - 6\sigma^2 E\{Y_1 Y_3\}) \right]. \quad (50)$$

In the above equation  $E\{Y_1^2\}$ , and  $E\{Y_3^2\}$  are respectively given by (36) and (42); therefore one needs to find  $E\{Y_1 Y_3\}$  whose value can be easily computed by combining (34) and (38) to give

$$\begin{aligned} E\{Y_1 Y_3\} &= \frac{1}{n^2} E \left\{ \sum_{i=1}^n W_i \sum_{j=1}^n (W_j + 3x_j^2 W_j + 3x_j W_j^2 - 3x_j \sigma_w^2) \right\} \\ &= (1 + \sigma_w^2) \frac{3\sigma_w^2}{n}. \end{aligned} \quad (51)$$

We can make several observations based on (50). For large amounts of noise we see from (48) that

$$\sigma \rightarrow 0.$$

Hence from (47)

$$E\{S\} = \sigma^3 \bar{s} + \sigma^3 E\{Y_3\} - 3\sigma E\{Y_1\} \rightarrow 0. \quad (52)$$

Therefore we can state the following lemma.

**Lemma 1:** Additive noise tends to move the expected value of edge location to the center.

Next, for a low signal-to-noise ratio, and for large  $|\bar{s}|$  values (when the edge location is toward the ends), the first two terms on the right-hand side of (50), which we call the "bias component," are dominating. On the other hand, for small values of  $|\bar{s}|$  (when the edge is close to the center), the last three terms which we call the "jitter components," are dominating. Fig. 4 provides an example for verification of the above statements. In this figure an input edge data of sequence length 20 was chosen. The x-axis corresponds to edge location and y-axis corresponds to rms error in edge location. The input signal-to-noise ratio is defined as

$$S/N \triangleq 10 \log \frac{\Delta h^2}{\sigma_w^2} \text{ dB} \quad (53)$$

where

$\Delta h$  = height difference of ideal edge

$\sigma_w^2$  = variance of the noise.

The signal-to-noise ratio was 6 dB for data in Fig. 4. The full line corresponds to the bias term, and the dashed line corresponds to the jitter terms. Furthermore, Fig. 5 compares empirical and theoretical rms edge location error.

## VI. PREPROCESSING OF DATA TO ACHIEVE MONOTONICITY

In the previous section, we calculated the edge location from observed data that we assumed to be corrupted by additive noise. However, in the presence of additive noise, application of some preprocessing techniques, such as averaging and/or median filtering [15] can on the average appreciably decrease the error incurred in calculating the edge location.

For low noise conditions, median filters have the property of removing impulses and oscillations while preserving monotonic edges [23]. On the other hand, the averaging operation tends to introduce distortion on edge location. To see that, let us assume we have an input edge data sequence  $x_m$ ,  $m = 1, 2, \dots, n$  defined as follows:

$$x_m = \begin{cases} 0 & m \leq r \\ 1 & m > r \end{cases} \quad (54)$$

If we use a three pixel wide averaging window the input and output sample moments  $\bar{m}_i$ , and  $\hat{m}_i$ ,  $i = 1, 2, 3$ , are related in the following way:

$$\hat{m}_1 = \bar{m}_1 \quad (55)$$

$$\hat{m}_2 = \bar{m}_2 - \frac{4}{9n} \quad (56)$$

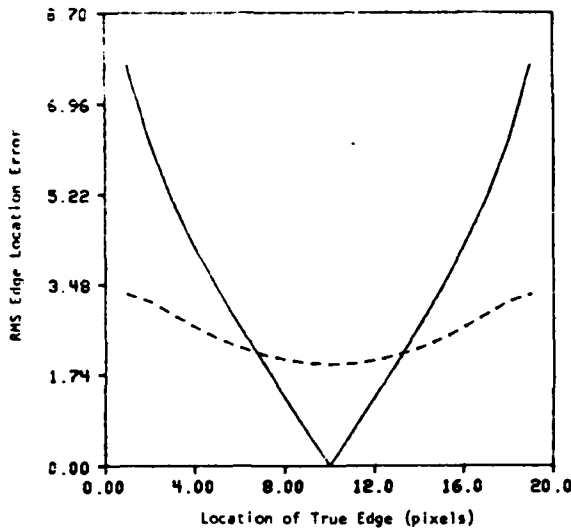


Fig. 4. Two components of edge location error for (50). The solid line represents the first two terms of (50) which are called "bias" (dc offset error due to the fact that additive noise tends to force the edge location toward the center of the data). The dashed line represents the remaining three terms in (50) which are called "jitter" (ac errors due to additive noise). The sequence length is 20 points and the signal-to-noise ratio is 6 dB.

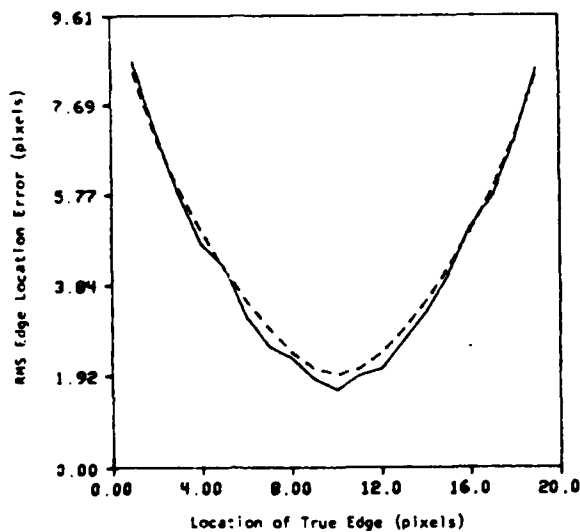


Fig. 5. Comparison of empirical and theoretical rms location errors. The dashed line represents the theoretical curve (50) which is the combination of the two components in Fig. 4 and the solid line shows the measured empirical error using 200 trials for each edge location. The sequence length is 20 points and the signal-to-noise ratio is 6 dB.

$$\hat{m}_3 = \bar{m}_3 - \frac{2}{3n} \quad (57)$$

and

$$\bar{m}_1 = \bar{m}_2 = \bar{m}_3 = \frac{n-r}{n} \quad (58)$$

The sample skewness associated with output sequence can then be written as

$$\hat{s} = \frac{\bar{m}_3 + 2\bar{m}_1^3 - 3\bar{m}_1\bar{m}_2}{\sigma^3} \quad (59)$$

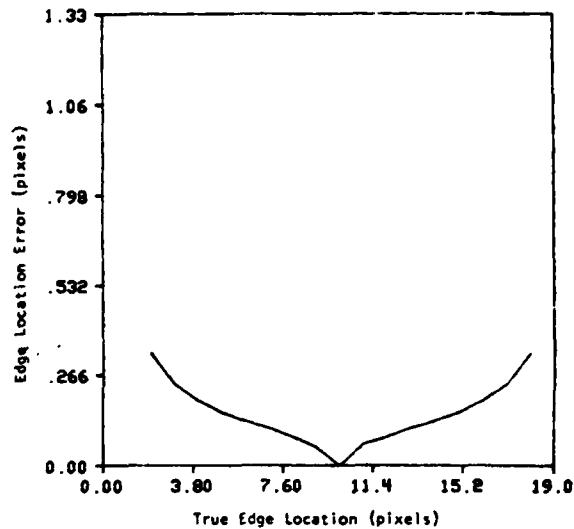


Fig. 6. Theoretical error introduced by averaging an ideal edge using a window of width three. The sequence length is 20 points.

Combination of (58)–(62) will then yield

$$\begin{aligned} \hat{s} &= \frac{\bar{m}_3 - \frac{2}{3n} + 2\bar{m}_1^3 - 3\bar{m}_1\left(\bar{m}_2 - \frac{4}{9n}\right)}{\left(\bar{\sigma}^2 - \frac{4}{9n}\right)^{3/2}} \\ &= \frac{\bar{\sigma}^3 \bar{s} + \frac{2}{3n} \left(\frac{n-2r}{n}\right)}{\left(\bar{\sigma}^2 - \frac{4}{9n}\right)^{3/2}} \end{aligned} \quad (60)$$

From (60) we can conclude that centered edges remain unchanged under the averaging operation. This is easily shown if one substitutes 0 for  $\bar{s}$  and  $n/2$  for  $r$  in (60). For noncentered edges when  $0 < r < n/2$ ,  $\bar{s}$  takes negative values, and for values of  $r$  in the range of  $n/2 < r < n$ ,  $\bar{s}$  takes positive values, therefore from (60) it can be easily seen that averaging tends to move the edge location toward the center, this is especially true if the assumption

$$\sqrt{\bar{\sigma}^2 - \frac{4}{9n}} \approx \bar{\sigma} \quad (61)$$

holds. Fig. 6 shows the effect of averaging on an ideal edge of sequence length 20, the x-axis corresponds to edge location, and y-axis corresponds to rms edge location error. For comparison purpose Fig. 7 shows rms edge location error versus various signal-to-noise ratio for a centered ideal edge of sequence length 20. Even though averaging tends to introduce some distortion, as a rule (especially for low signal-to-noise ratios) one can use an averaging filter in the presence of noise as a means to reduce error in edge location estimation. The reason why this is true can be explained in the following way:

$$Z_i = x_i + W_i \quad i = 1, 2, \dots, n. \quad (62)$$

Then, if a "k + 1" size averaging window is used, we will have

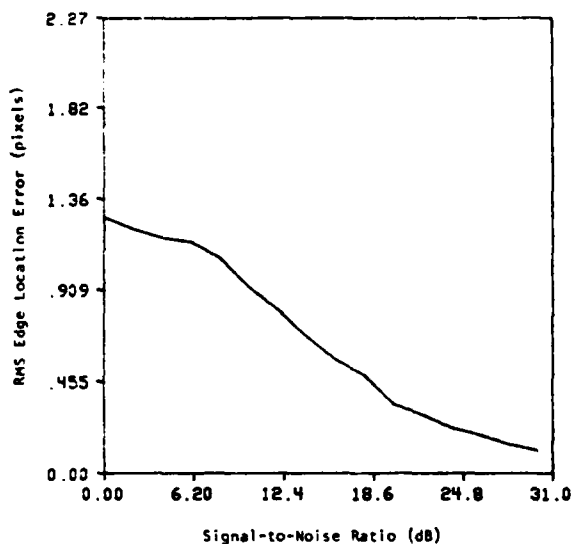


Fig. 7. Rms edge location error as a function of signal-to-noise ratio. The sequence length is 20 and the edge is located at the center of the sequence.

$$\begin{aligned}\hat{Z}_i &= \frac{1}{k+1} \sum_{j=i-k/2}^{i+k/2} Z_j \\ &= \frac{1}{k+1} \sum_{j=i-k/2}^{i+k/2} x_j + \frac{1}{k+1} \sum_{j=i-k/2}^{i+k/2} w_j \\ &= \hat{x}_i + \hat{w}_i.\end{aligned}\quad (63)$$

From (63) the mean and variance of  $\hat{w}_i$  can be easily obtained and are equal to

$$E\{\hat{w}_i\} = 0 \quad (64)$$

$$\text{var}\{\hat{w}_i\} = \frac{\sigma_w^2}{k+1}. \quad (65)$$

As can be seen from the above equation, averaging always lowers the variance of noise by a factor of " $k+1$ " which is equal to the width of the averaging filter. But as it can be seen from Table II, there should exist a thresholding signal-to-noise ratio such that above that value use of a median filter is advisable. Pomalaza [24] has compared the performance of a 3-pixel wide median filter with an averaging filter of the same size when applied to an ideal edge of two adjacent intensities in the presence of additive uncorrelated noise. He has calculated the variance of the output of the median filter around the discontinuity and has concluded that the height of the discontinuity has to be almost twice the standard deviation of the noise for the running median estimate to start having less mean-square error than the running mean estimate.

## VII. EXTENSION TO TWO-DIMENSIONAL EDGE PATTERNS

In this section, the one-dimensional edge operator discussed previously is extended so that it can operate on two-dimensional edge patterns. This technique identifies an edge location by a line equation whose two parameters are calculated according to criteria to be discussed in later sections. Also in

TABLE II  
COMPUTED MEAN ABSOLUTE ERROR MEASURES WITH AND WITHOUT  
PREPROCESSING. SEQUENCE LENGTH = 20, EDGE LOCATION = 6,  
NUMBER OF TRIALS = 200

Mean Absolute Error	
For S/N = 0dB	
No preprocessing	4.723
Using median filter only	4.040
Using averaging only	3.863
Using median filter first and then averaging	4.030
Averaging first and then using median filter	3.857
For S/N = 6dB	
No preprocessing	3.715
Using median filter only	2.759
Using averaging only	2.410
Using median filter first and then averaging	2.503
Averaging first and then using median filter	2.074
For S/N = 10dB	
No preprocessing	2.188
Using median filter only	1.411
Using averaging only	1.153
Using median filter first and then averaging	1.138
Averaging first and then using median filter	1.033
For S/N = 20dB	
No preprocessing	.306
Using median filter only	.154
Using averaging only	.175
Using median filter first and then averaging	.176
Averaging first and then using median filter	.168

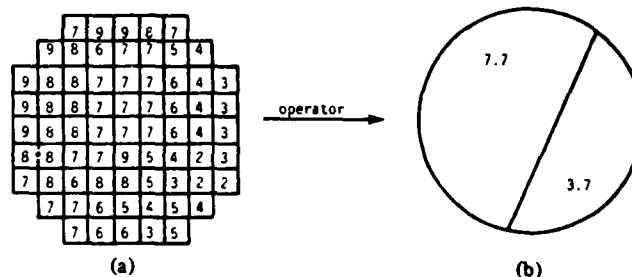
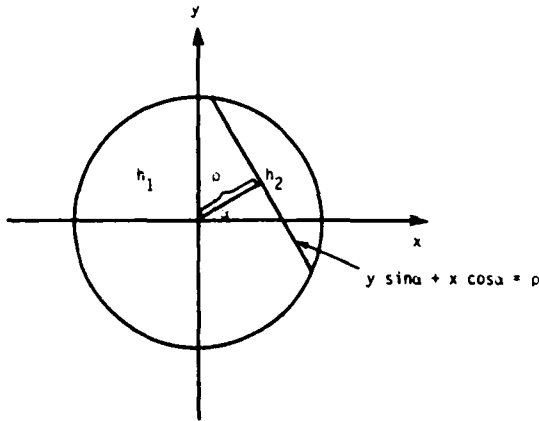


Fig. 8. (a) Empirically obtained edge element. (b) Ideal edge element.

this section the two-dimensional edge operator is compared with Hueckel's edge operator.

In implementing the edge operator, an approach similar to Hueckel [4] is taken to define the input and output of the operator. In particular, the edge operator accepts as input a set of grid squares consisting of 69 pixels, arranged so as to best approximate the area of a unit circle [see Fig. 8(a)]. As an output, the edge operator generates an ideal edge element defined over a unit circle with two brightness values  $h_1$  and  $h_2$ , along with the borderline that separates the two intensity levels as shown in Figs. 8(b) and 9. A more exact definition

Fig. 9. Edge line equation as a function of  $\alpha$  and  $\rho$ .

of the output disk is obtained if we denote the ideal edge element by  $U(x, y, \sin \alpha, \cos \alpha, \rho, h_1, h_2)$ , then

$$U(x, y, \sin \alpha, \cos \alpha, \rho, h_1, h_2) = h_1 \quad \text{if } y \sin \alpha + x \cos \alpha \leq \rho$$

$$U(x, y, \sin \alpha, \cos \alpha, \rho, h_1, h_2) = h_2 \quad \text{if } y \sin \alpha + x \cos \alpha > \rho.$$

(66)

However, Hueckel chooses the quintuples  $(\sin \alpha, \cos \alpha, \rho, h_1, h_2)$  so as to minimize the  $L_2$  norm between input and output data, i.e., he minimizes the quantity  $\Delta(\cdot)$ ,

$$\Delta(\cdot) = \int_{\text{unit}} \int_{\text{O}} [I(x, y) - U(x, y, \sin \alpha, \cos \alpha, \rho, h_1, h_2)]^2 dx dy \quad (67)$$

where  $I(x, y)$  is continuous and constant over each grid associated with input data.

In our technique we first find the quantities  $\rho, h_1$ , and  $h_2$  by preserving the first three sample moments between input and output of the operator in the following manner:

$$\bar{m}_k \triangleq \frac{1}{\pi} \iint_D I^k(x, y) dx dy = h_1^k p_1 + h_2^k p_2 \quad k = 0, 1, 2, 3 \quad (68)$$

where

$$p_1 = \frac{A_1}{\pi} \quad (69)$$

$$A_1 \triangleq \text{the area on } D \text{ covered by intensity } h_1 \quad (70)$$

$$p_2 = 1 - p_1 \quad (71)$$

$$D = \{x, y: x^2 + y^2 \leq 1\}. \quad (72)$$

By assuming that  $I(x, y)$  takes constant values over each grid, then the integral in (68) becomes a weighted sum of intensities in the input disk. Therefore one can write

$$\bar{m}_k = \sum_{j=1}^{69} w_j I_j^k \quad k = 0, 1, 2, 3 \quad (73)$$

$I_j$  = intensity associated with  $j$ th grid

$w_j$  = weighting associated with  $j$ th grid.

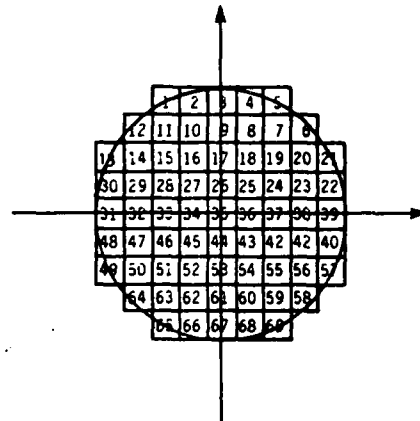


Fig. 10. Indexing associated with each square grid.

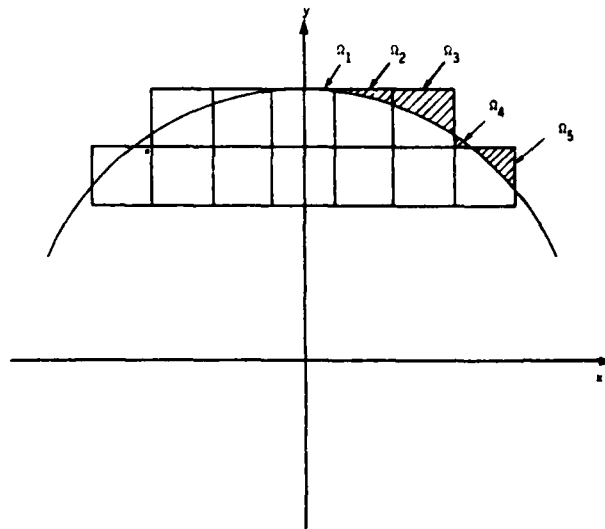


Fig. 11. Areas to be added or subtracted for the calculation of weighting coefficients.

One should note that indexes in (73) refer to grids shown in Fig. 10, each having an area "A" equal to

$$A = 4d^2 \quad (74)$$

where

$$d = \frac{1}{9}.$$

The value "1/9" was chosen because of the requirement that the input disk should best approximate a circle of radius one.

To find the weightings associated with each grid intensity, if one takes into account the symmetry that exists among grid areas, then

$$\begin{aligned} w_1 &= w_5 = w_{21} = w_{57} = w_{69} = w_{65} = w_{49} = w_{13} \\ &= \frac{1}{\pi} \left( A - \iint_{\Omega_1} dx dy + \frac{1}{2} \iint_{\Omega_4} dx dy \right) \\ &= 8.4670539 \times 10^{-3} \end{aligned} \quad (75)$$

where  $\Omega_3$  and  $\Omega_4$  correspond to the area shown in Fig. 11. The last term is added to take into account the areas of the circle that are not covered by the grid. Similarly,

$$w_2 = w_4 = w_{22} = w_{40} = w_{68} = w_{66} = w_{48} = w_{30} = 0.13782918 \quad (76)$$

$$w_3 = w_{31} = w_{39} = w_{67} = 0.015573185 \quad (77)$$

Finally,

$$w_6 = w_{58} = w_{64} = w_{12} = 0.013068037. \quad (78)$$

The remaining weighting coefficients are assigned a value of "0.015719006."

Once the moments  $\bar{m}_1$ ,  $\bar{m}_2$ , and  $\bar{m}_3$  are obtained, then (68) corresponds to (5), and, hence the values of  $p_1$ ,  $p_2$ ,  $h_1$ ,  $h_2$  can easily be determined from (6)-(8).

Given a circle of radius unity, and an arbitrary angle  $0 \leq \beta < \pi/2$ , then the area "A" shown in Fig. 12 is given by

$$A = \beta - \frac{1}{2} \sin 2\beta. \quad (79)$$

If we let

$$p = \min(p_1, p_2) \quad (80)$$

then, by combining (60) and (79), one can write

$$\beta - \frac{1}{2} \sin 2\beta = \pi p \quad (81)$$

or

$$\beta - \beta \frac{\sin \beta}{\beta} \cos \beta = \pi p. \quad (82)$$

Equation (82) is a transcendental equation, and one can use some numerical approximations [19] in order to obtain  $\beta$ . Once  $\beta$  is obtained then

$$\rho = \cos \beta \quad (83)$$

and, hence, by preserving the first three sample moments, one is able to determine  $h_1$ ,  $h_2$ , and  $\rho$ .

Still left to be found is the direction of the edge, or in other words, the slope of the border line separating the two intensities  $h_1$  and  $h_2$ . The following approach can be used to find the direction of the edge.

Assume an ideal edge element with grey level intensities  $h_1$  and  $h_2$  inside the circle  $x^2 + y^2 = 1$ . Define

$$\bar{x} = \frac{h_1 \int_{\Omega_1} x \, dx \, dy + h_2 \int_{\Omega_2} x \, dx \, dy}{h_1 \int_{\Omega_1} dx \, dy + h_2 \int_{\Omega_2} dx \, dy} \quad (84)$$

$$\bar{y} = \frac{h_1 \int_{\Omega_1} y \, dx \, dy + h_2 \int_{\Omega_2} y \, dx \, dy}{h_1 \int_{\Omega_1} dx \, dy + h_2 \int_{\Omega_2} dx \, dy} \quad (85)$$

as the coordinates of the center of gravity of intensities inside the circle, then the direction of the edge is perpendicular to the direction of the vector from the origin to  $(\bar{x}, \bar{y})$ . Machuca and Gilbert [9] have used the above idea in deriving their edge detector. Therefore, by combining (7), (8), (84), and (85), one can write the edge line equation as

$$\begin{aligned} y \sin \alpha + x \cos \alpha &= -\rho & p_1 < p_2 \\ y \sin \alpha + x \cos \alpha &= \rho & p_1 > p_2 \end{aligned} \quad (86)$$

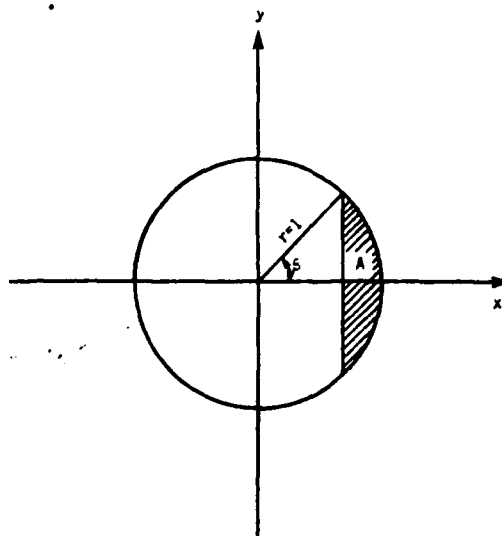


Fig. 12. Region A shows the area to be calculated in (79).

where

$$\sin \alpha = \frac{\bar{y}}{\sqrt{\bar{x}^2 + \bar{y}^2}}$$

$$\cos \alpha = \frac{\bar{x}}{\sqrt{\bar{x}^2 + \bar{y}^2}}$$

and

$$\bar{x} = \frac{\sum_{j=1}^{69} x_j I_j}{\sum_{j=1}^{69} I_j}$$

$$\bar{y} = \frac{\sum_{j=1}^{69} y_j I_j}{\sum_{j=1}^{69} I_j}$$

$I_j \triangleq$  intensity of  $j$ th grid

$(x_j, y_j) \triangleq$  coordinates of the center of  $j$ th grid.

A minus sign appears on the right-hand side of (86) when  $p_1 < p_2$ , because the center of gravity should always be located closer to the set of pixels with higher intensities.

Table III shows the results obtained when the operator is applied to different empirical input edge patterns. Also, for the sake of comparison, the results obtained when the Hueckel edge operator is applied to the same input edge patterns, are shown.

#### VIII. THE EFFECT OF ADDITIVE NOISE ON 2-D EDGE OPERATOR

In Section V an analysis of the effect of an additive white Gaussian noise on  $p$  was provided. The effect of additive noise on the slope of the edge line can also be theoretically analyzed [19]. Here we will present only empirical results.

Figs. 13 and 14 show the effect of noise ( $S/N = 20$  dB) on





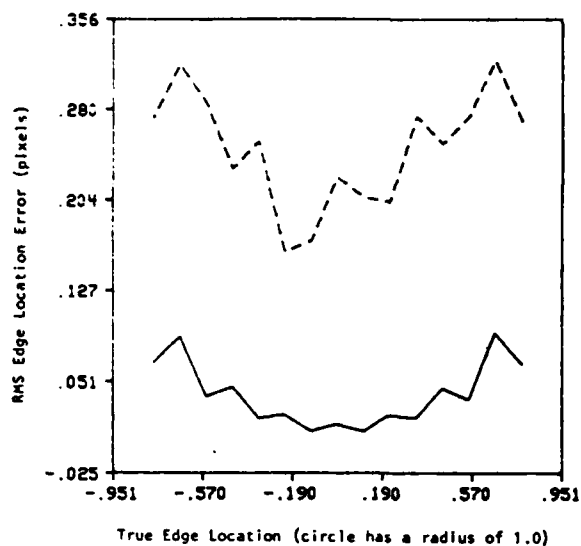


Fig. 13. Effect of additive noise on quantity  $\rho$  of edge line equation for the proposed 2-D edge operator (solid line) and the Hueckel edge operator (dashed line). The signal-to-noise ratio is 20 dB.

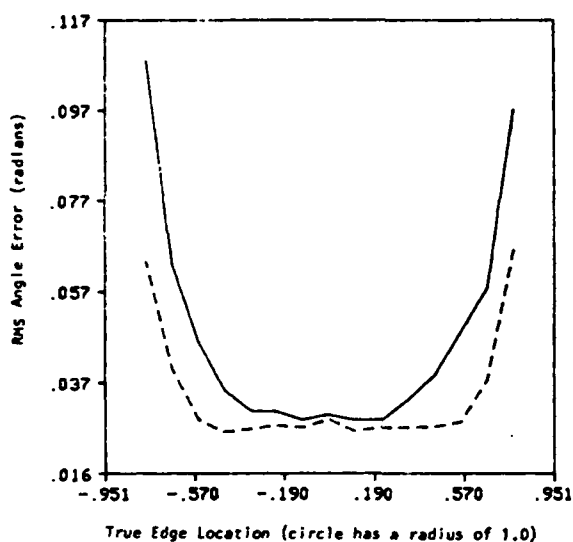


Fig. 13. Effect of additive noise on the slope of the edge line equation for the proposed 2-D edge operator (solid line) and the Hueckel edge operator (dashed line). The signal-to-noise ratio is 20 dB.

the slope and on the quantity " $\rho$ " for both the new edge and the Hueckel edge operators. As it can be seen from the figures, the new edge operator outperforms the Hueckel edge operator when estimating  $\rho$  and does almost as well on estimating the slope of the line. Note that the solid line corresponds to the new edge operator and the dashed line corresponds to the Hueckel edge operator.

#### IX. APPLICATION OF EDGE OPERATOR AS AN EDGE DETECTOR

In this section, our edge operator was repeatedly applied to a digital picture of size  $256 \times 256$  (see Fig. 15). The size of the input disk used was that of Fig. 8, and whenever an ac-

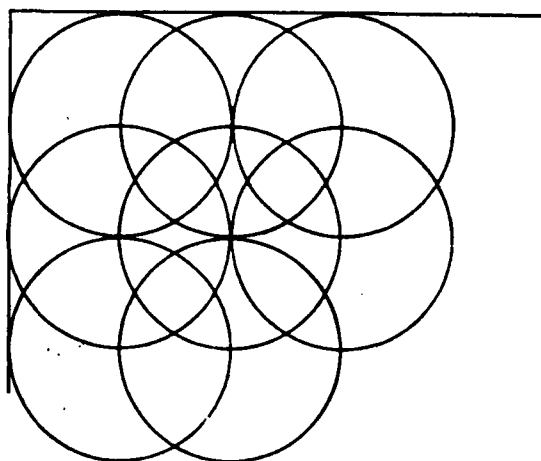


Fig. 15. The location of edge operators for the purpose of edge detection.

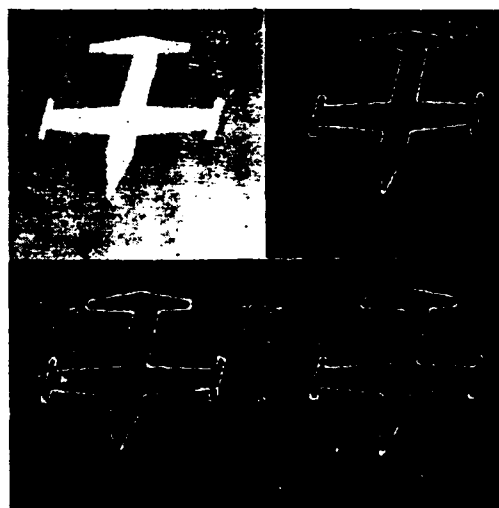


Fig. 16. Results of applying various edge detectors to a binary airplane image. Upper left: Original range ( $256 \times 256$  pixels). Upper right: Sobel edge detector. Lower left: Hueckel edge detector. Lower right: New 2-D edge detector.

ceptable edge pattern was encountered on the input disk the edge operator would generate the edge line equation as output. For the purpose of display the edge line equation was used to set pixels closest to the edge line to some predetermined intensity value (e.g., 255) resulting in a binary output picture.

Fig. 16 shows the original picture, along with the result of the application of our edge detector with that of Hueckel and Sobel. Fig. 17 shows the performance of different edge operators in the presence of noise (signal-to-noise ratio is 6 dB).

The criteria for acceptance of a pattern as an edge was based on

$$|h_1 - h_2| \geq \lambda \quad (87)$$

where

$$|h_1 - h_2| = \text{magnitude of height difference for the projected ideal edge.}$$

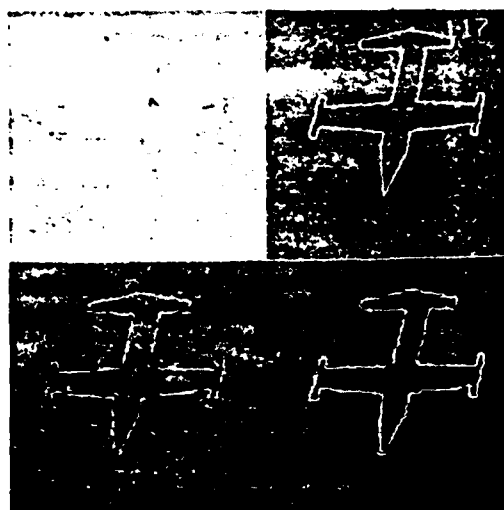


Fig. 17. Results of applying various edge detector to a binary airplane image with additive noise (SNR = 6 dB). Upper left: Original image. Upper right: Sobel edge detector. Lower left: Hueckel edge detector. Lower right: New 2-D edge detector.

A lower bound on the value of  $\lambda$  can be found in the following way. From (6) and (7) we have

$$\frac{(h_1 - h_2)^2}{\sigma^2} = \frac{1}{p_1 p_2} \quad (88)$$

Since  $p_1 + p_2 = 1$ , and  $p_1 p_2 \leq \frac{1}{4}$  therefore

$$(h_1 - h_2)^2 \geq 4\sigma^2 \quad (89)$$

or

$$|h_1 - h_2| \geq 2\sigma \quad (90)$$

where

$$\sigma^2 \triangleq \text{variance of the observed data.}$$

Now, if (88) is not satisfied then we conclude that an edge pattern is not present, otherwise we test for the condition

$$\left| \frac{d_i - d_e}{d_i + d_e} \right| \leq 0.5 \quad (91)$$

where

$$d_i = \frac{2(h_1 - h_2) \sin^3 s}{3m_1 \pi}$$

= distance of the origin from the center of gravity of projected ideal edge

$$d_e = \sqrt{\bar{x}^2 + \bar{y}^2}$$

= distance of the origin from the center of gravity of empirical input edge pattern.

If condition (91) is satisfied as well as (88), then we conclude that an edge pattern is present.

## X. CONCLUSIONS AND DISCUSSION

A new edge operator has been presented which can locate an edge to subpixel accuracy in one- and two-dimensional data.

The method gives the edge location in closed form and requires no interpolation or iteration. It is invariant to additive and multiplicative grey level changes. The method assures that the data are monotonic over each segment and thus each edge region must be first detected before the exact location is measured. If noise is present in the data, it is often beneficial to preprocess the data (averaging or median filtering) prior to the edge location operation. It is noted that noise in general tends to bias the detected edge location toward the center of the data. Therefore the most accurate edge location can be made when the edge is as near to the center of the data segment as possible.

Although the operator was developed to locate edges, it may be used to locate other shapes. For example, a pulse shape in one-dimensional data may be located by first integrating the data and then locating the edge in the resulting sequence. A thin line may be located as two-dimensional data similarly by performing a two-dimensional integration and locating the resulting two-dimensional edge. Other shapes may be located by first preprocessing (filtering to remove unwanted frequencies), integrating (to give a monotonic sequence), and fitting a parametric monotonic curve by matching moments as described in Section IV.

## REFERENCES

- [1] E. M. Mikhail and M. M. Thompson, "Automation in photogrammetry: Recent developments 1972-1976" (invited paper), presented at the 13th Cong. Int. Soc. Photogrammetry, Helsinki, Finland, July 1976.
- [2] M. Mikhail, J. C. McGlone, and J. R. Baker, "Metric information from aircraft multispectral scanner (MSS) data," presented at the 1979 Amer. Soc. Photogrammetry Spring Convention, Washington, DC, Mar. 1979.
- [3] M. F. Hueckel, "An operator which locates edges in digitized pictures," *J. Ass. Comput. Mach.*, vol. 18, pp. 113-125, 1971.
- [4] —, "A local operator which recognizes edges and lines," *J. Ass. Comput. Mach.*, vol. 20, pp. 634-647, 1973.
- [5] —, "Erratum for [4]," *J. Ass. Comput. Mach.*, vol. 21, p. 350, Apr. 1974.
- [6] W. Frei and C. C. Chen, "Fast boundary detection: A generalization and a new algorithm," *IEEE Trans. Comput.*, vol. C-26, pp. 988-998, Oct. 1977.
- [7] L. G. Roberts, "Machine perception of three-dimensional solids," in *Optical and Electro-Optical Information Processing*, J. T. Tippett et al., Eds. Cambridge, MA: MIT Press, 1965.
- [8] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*. New York: Wiley, 1971.
- [9] R. Machuca and A. L. Gilbert, "Finding edges in noisy scenes," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, pp. 103-111, Jan. 1981.
- [10] B. Schuster and A. Rosenfeld, "Some new methods of detecting step edges in digital pictures," *Commun. Ass. Comput. Mach.*, vol. 21, pp. 172-176, Feb. 1978.
- [11] J. Burnett and T. S. Huang, "Image mensuration by maximum a posteriori probability estimation," *J. Opt. Soc. Amer.*, Feb. 1978.
- [12] C. J. Jacobus and R. T. Chien, "Two new edge detectors," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-3, pp. 581-592, Sept. 1981.
- [13] A. Rosenfeld and A. Kak, *Digital Picture Processing*, 2nd ed. New York: Academic, 1982.
- [14] L. S. Davis, "A survey of edge detection techniques," *Comput. Graphics Image Processing*, vol. 4, p. 248, Sept. 1975.
- [15] W. Pratt, *Digital Image Processing*. New York: Wiley, 1978.
- [16] I. Abdou, *Quantitative Methods of Edge Detection*, Los Angeles, CA: Image Processing Inst., Univ. Southern California, 1978.
- [17] E. J. Delp and O. R. Mitchell, "Image compression using block truncation coding," *IEEE Trans. Commun.*, vol. COM-27, pp. 1335-1341, Sept. 1979.

- [18] G. Szego, *Orthogonal Polynomials*, vol. 23, 4th ed. Providence, RI: Amer. Math. Soc., 1975.
- [19] A. Tabatabai, "Edge location and data compression for digital imagery," Ph.D. dissertation, School Elec. Eng., Purdue Univ., Dec. 1981.
- [20] T. W. Hill, "On determining a distribution function known only by its moments and/or moment generating function," Ph.D. dissertation, Arizona State Univ.
- [21] R. Von Mises, *A Mathematical Theory of Probability and Statistics*. New York: Academic, 1964.
- [22] W. P. Elderton and V. L. Johnson, *Systems of Frequency Curves*. London-New York: Cambridge Univ. Press, 1969.
- [23] N. C. Gallagher and Wise, "A theoretical analysis of the properties of median filters," *IEEE Trans. Acoust., Speech, Signal Processing*, to be published.
- [24] C. A. Pomalaza, "Improved and robust estimation of nonrandom waveforms," Ph.D. dissertation, Purdue Univ., Aug. 1980.

communication theory and digital image processing (coding and pattern recognition).

Dr. Tabatabai is a member of Eta Kappa Nu.



O. Robert Mitchell (S'64-M'72-SM'82) was born in Beaumont, TX, on July 4, 1945. He received the B.S.E.E. degree from Lamar University, Beaumont, TX, in 1967 and the S.M.E.E. and the Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, in 1968 and 1972, respectively.

From 1969 to 1971, he was a Software Engineer designing optical character recognition equipment at ECRM, Inc., Cambridge, MA.

In 1972 he joined the faculty of Purdue University, West Lafayette, IN, where he is presently Associate Professor of Electrical Engineering. He spent the summers of 1977 and 1979 at White Sands Missile Range, where he familiarized himself with the problems of real-time video tracking and pattern recognition. While at WSMR, he developed several texture and shape description algorithms. He is active in teaching, research, and consulting in the areas of image coding, texture analysis, shape description, pattern recognition, and signal processing.

Dr. Mitchell is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi.



Ali J. Tabatabai was born in Paris, France, on June 20, 1950. He received the B.S.E.E. degree from Tohoku University, Japan, in 1974, the M.S.E. degree from Arizona State University, Tempe, in 1976, and the Ph.D. degree from Purdue University, West Lafayette, IN, in 1981.

Since September 1981 he has been with Bell Laboratories, Holmdel, NJ, although he spent some time working for American Bell in the area of videotex. His main interests are in

APPENDIX C

A MOMENT BASED TWO-DIMENSIONAL EDGE OPERATOR

A. P. Reeves, M. L. Akey, and O. R. Mitchell

IEEE Computer Society Conference on Computer Vision  
and Pattern Recognition

Washington, D. C.

June 19-23, 1983

## A MOMENT BASED TWO-DIMENSIONAL EDGE OPERATOR

Anthony P. Reeves  
School of Electrical Engineering  
Cornell University  
Ithaca, NY 14853

Mark L. Akey and O. Robert Mitchell  
School of Electrical Engineering  
Purdue University  
West Lafayette, IN 47907

### Abstract

A two-dimensional edge operator is developed which matches an ideal step edge to a window of data using two-dimensional moments. This method requires no iteration and can locate edges to sub-pixel accuracy. Sensitivity of the operator to noise is evaluated both theoretically and empirically.

### I. Introduction

It is often necessary to find straight edges in a digital image and to locate their position precisely. Many edge detection schemes have been proposed [1-3] which are based on gradient methods (such as Sobel) or template correlation (such as Frei and Chen). Edge location is more difficult, the most accepted method being that proposed by Hueckel [4-6].

An edge detector and locator is proposed here which matches a circular section of an image to an ideal step edge model using two-dimensional moments. This method is much simpler to implement than that of Hueckel and appears to allow more accuracy and noise immunity.

### II. Definition of the Edge Operator

The ideal edge model is shown in Fig. 1 and is characterized by four parameters  $h$ ,  $k$ ,  $\theta$  and  $\phi$ . The edge is a straight line which separates two regions of constant grey values. The lower level has height  $h$  and the upper level is  $k$  higher than the lower level. The angle which the edge makes with the  $y$  axis is  $\theta$  and  $\phi$  is the distance from the center of the disk to the edge.

The moments of an image  $f(x,y)$  of order  $p+q$  are defined by

$$M_{pq} = \int \int x^p y^q f(x,y) dx dy \quad (1)$$

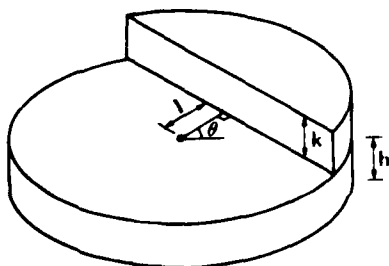


Fig. 1. Model of an ideal edge.

The disk is defined to have radius of one. Thus the limits of integration are the unit circle i.e.,  $\sqrt{x^2+y^2} \leq 1$ . A closed moment set (CMS) of order  $n$  consists of all moments of order  $n$  and lower and is closed with respect to the operations rotation, translation and scale change. For the edge detector the CMS of order-2 is computed i.e.  $\{M_{00}, M_{01}, M_{10}, M_{02}, M_{11}, M_{20}\}$ .

A rotation of the disk by an angle  $\phi$  changes the moments as specified by

$$M'_{pq} = \sum_{r=0}^p \sum_{s=0}^q \begin{pmatrix} p \\ r \end{pmatrix} \begin{pmatrix} q \\ s \end{pmatrix} (-1)^{r+s} (\cos \phi)^{p-r} (\sin \phi)^{q-s} M_{r+s, p+q-r-s} \quad (2)$$

First, to obtain  $\theta$ , consider rotating the edge region so that the edge is aligned with the  $y$  axis as shown in Fig. 2a. At this position there is symmetry about the  $x$  axis therefore

$$M'_{01} = 0 \quad (3)$$

From (2) the value of  $M'_{01}$  can be obtained in terms of  $\theta$ .

$$M'_{01} = M_{01} \cos \theta - M_{10} \sin \theta \quad (4)$$

From (3) and (4)  $\theta$  is determined

$$\theta = \tan^{-1} \frac{M_{01}}{M_{10}} \quad (5)$$

In order to determine the other edge model parameter, the moment set is rotated by the angle  $\theta$  using (2) until the potential edge is aligned with the  $y$  axis as shown in Fig. 2a. The value derived for  $\theta$  from (5) may need a correction of  $\pi$  since there are two possible ways to align an edge with the  $y$  axis. A unique value of  $\theta$  is obtained by the additional constraint that  $M'_{10} > 0$  this ensures that the higher level of the rotated edge is on the right and the lower level is on the left in Fig. 2a.

The location of the edge,  $\phi$ , may be derived from the rotated CMS  $\{M'_{pq}\}$ . In fact, only the moments with respect to the  $x$  axis i.e.  $\{M'_{00}, M'_{10}, M'_{20}\}$  are required. This

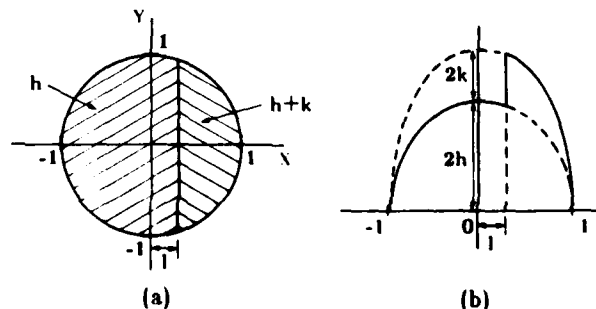


Fig. 2. a) Ideal edge aligned with the  $y$ -axis, b) Projection of edge onto the  $x$ -axis.

moment set corresponds to the moments of a projection through the edge model onto the x axis as shown in Fig. 2b. The moments may be specified in terms of h, k and  $\theta$  by integrating of the elliptical shapes shown in Fig. 2b.

$$M'_{00} = 2h \int_{-1}^1 \sqrt{1-x^2} dx + 2k \int_{-1}^1 \sqrt{1-x^2} dx$$

$$= h\pi + \frac{k\pi}{2} - k \sin^{-1}(\theta) - k\theta\sqrt{1-\theta^2} \quad (6)$$

$$M'_{10} = 2h \int_{-1}^1 x\sqrt{1-x^2} dx + 2k \int_{-1}^1 x\sqrt{1-x^2} dx$$

$$= \frac{2}{3} k \sqrt{(1-\theta^2)^3} \quad (7)$$

$$M'_{20} = 2h \int_{-1}^1 x^2 \sqrt{1-x^2} dx + 2k \int_{-1}^1 x^2 \sqrt{1-x^2} dx$$

$$= \frac{h\pi}{4} + \frac{k\pi}{8} + \frac{k\theta}{2} \sqrt{(1-\theta^2)^3}$$

$$- \frac{k}{4} (\theta\sqrt{1-\theta^2} + \sin^{-1}(\theta)) \quad (8)$$

Equations (6), (7) and (8) may now be combined to solve for  $\theta$  i.e.

$$M'_{20} = \frac{3}{4} \theta M'_{10} - \frac{1}{4} M'_{00} \quad (9)$$

$$\theta = \frac{4M'_{20} - M'_{00}}{3M'_{10}} \quad (10)$$

Once  $\theta$  is determined the two values h and k may be obtained by back substitution

$$k = \frac{3 M'_{10}}{2\sqrt{(1-\theta^2)^3}} \quad (11)$$

$$h = \frac{2M'_{00} - k(\pi - 2\sin^{-1}(\theta) - 2\theta\sqrt{1-\theta^2})}{2\pi} \quad (12)$$

The other moments in the CMS, i.e.  $\{M'_{01}, M'_{11}, M'_{02}\}$  may also be specified with respect to h,  $\theta$  and K. Since there is symmetry about the x axis in Fig. 2a,

$$M'_{01} = 0, \quad (13)$$

$$M'_{11} = 0. \quad (14)$$

To determine  $M'_{02}$  consider the projection of Fig. 2a split into two components, if  $\theta \geq 0$ . (When  $\theta < 0$  compute  $\frac{k\pi}{4}$  - the section of the k height disk).

$$M'_{02} = 2h \int_{-1}^1 y^2 \sqrt{1-y^2} dy + k \int_{-1}^1 y^2 \sqrt{1-y^2} dy$$

$$- k \int_{-1}^1 y^2 \theta dy$$

$$= \frac{h\pi}{4} + \frac{k}{4} (\theta\sqrt{1-\theta^2} + \sin^{-1}(\theta\sqrt{1-\theta^2}))$$

$$- \frac{\sqrt{1-\theta^2}\theta^3}{2} - \frac{2k\theta}{3} \sqrt{(1-\theta^2)^3}$$

$$= \frac{1}{4} M'_{00} - \frac{k\theta\sqrt{1-\theta^2}}{6} (3\sqrt{\theta} + 1 + 4\theta^2) \quad (15)$$

Once  $\theta$  and k have been obtained for a potential edge the value for  $M'_{02}$  can be predicted. A figure of confidence can then be generated by comparing the predicted and actual values for  $M'_{02}$ . Alternatively, the value of  $M_{11}$  can also be used as a confidence measure.

### III. Implementation of the Edge Operator

In order to compute the parameters, the 6 values of the CMS of order 2 must be estimated then the model parameters can be computed. Each estimate is made by multiplying the elements in the local area of a pixel with a weight mask and summing the results. First,  $M_{01}$  and  $M_{10}$  are needed to estimate  $\theta$  by (5). Then to obtain the rotated moments, use

$$\cos(\theta) = \frac{M_{10}}{M_a}$$

$$\sin(\theta) = \frac{M_{01}}{M_a} \quad (16)$$

where

$$M_a = \sqrt{M_{01}^2 + M_{10}^2} \quad (17)$$

These values can be used to obtain the rotated moments by substituting into (2).

$$M'_{00} = M_{00}$$

$$M'_{10} = M_a$$

$$M'_{20} = \frac{(M_{10}^2 M_{20} + 2M_{01} M_{10} M_{11} + M_{01}^2 M_{02})}{M_a^2} \quad (18)$$

Then  $\theta$  is obtained by substituting these values into (10)

$$\theta = \frac{M_b}{M_a^2} - \frac{M_c}{M_a} \quad (19)$$

where

$$M_b = \frac{4}{3} (M_{10}^2 M_{20} + 2M_{01} M_{10} M_{11} + M_{01}^2 M_{02}) \quad (20)$$

$$M_c = \frac{1}{3} M_{00} \quad (21)$$

When needed the level height parameters can be computed from  $\{M_a, M_b, M_c\}$ . The procedure to obtain the edge parameters is as follows:

1. Estimate  $\{M_{00}, M_{01}, M_{10}, M_{02}, M_{11}, M_{20}\}$ .
2. Compute  $\theta$  from (5).
3. Compute  $M_a$  from (17).
4. Compute the distance to the edge,  $\theta$  from (19) through (21).
5. The edge height k may be obtained from (11).
6. The background level h may be obtained from (12).

### IV. Moment Value Estimation

In practice a region of an image is usually represented by a matrix of sampled intensity values called pixels. In order to estimate the moments for a circular image from a set of pixels, several assumptions and approximations must be made.

A circular region defined on a 5x5 matrix of pixels is shown in Fig. 3. The problem is to compute the exact contribution (weight) of each pixel to each moment value. The first assumption is that a pixel contains the mean value of square region it represents.

In Fig. 4 the three basic monomial functions are shown in one dimension. If a pixel has a constant value and is completely inside then the contribution to a moment function is proportional to the integral of the basis function in the region of the pixel. For a pixel located at point i with width j, the pixel region is  $i - \frac{1}{2}$  to  $i + \frac{1}{2}$ . For  $M_0$  we have

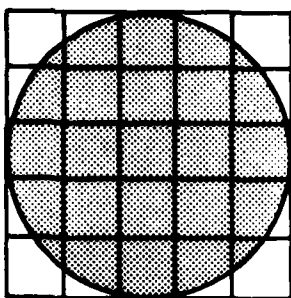


Fig. 3. A circular region defined on a 5 x 5 pixel matrix.

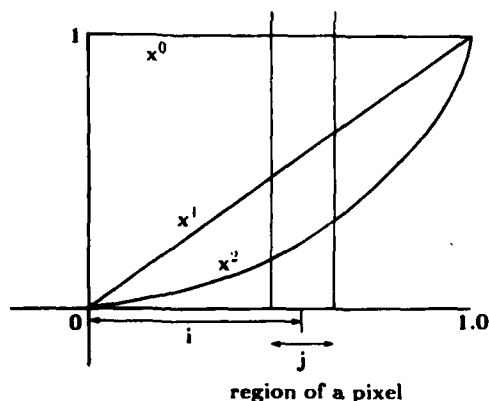


Fig. 4. Three basis monomials and a pixel region width  $j$ .

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} x^0 dx = j \quad (22)$$

Since  $j$  is constant for each pixel, the weight is the same for each pixel.

For  $M_1$  we have

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} x^1 dx = ij \quad (23)$$

In this case the weight of each pixel is proportional to its position. Weight masks for  $M_{01}$  and  $M_{10}$  can be generated using this algorithm. The product of these two masks gives the mask for  $M_{11}$  since it is separable.

Finally for  $M_2$  we have

$$\int_{-\frac{1}{2}}^{\frac{1}{2}} x^2 dx = i^2 j + \frac{1}{12} j^3 \quad (24)$$

In this case we get the expected  $i^2 j$  term which indicates that the position of the pixel squared is the dominant factor, however, there is also a small constant term which must also be added. Masks can be generated for  $M_{20}$  and  $M_{02}$  using (24).

Pixels whose regions are intersected by the boundary of the circle must be treated in a special way. Since we only have to generate these masks once, a simple brute force method was used to generate the weights for these pixels for our experiments. Each pixel region was split into a 401 x 401 submatrix; the contributions of each

subpixel within the circle to the moment value was then summed. Since the sub pixels cover a very small area, any sub pixels which were intersected by the circle boundary could be ignored without significantly affecting the total summed value for the whole pixel.

## V. Radial Weighting

Hueckel [3] has argued that the basis functions for an edge detector should diminish to zero at the edge of the disk. That is, more emphasis is placed on the pixels near the center of the disk, the idea being that extraneous, less-reliable information is more likely to be located near the edge of the disk. If this assumption is made, then the edge detector should only be used to detect edges near the center of the disk.

The moment basis functions, except for  $M_{00}$ , all have greater value at the edge of the disk than the center. Two radial weighting functions were considered for the moment edge detector. The first function  $\sqrt{1-r^2}$  where  $r = \sqrt{x^2 + y^2}$  is equivalent to multiplying the circular edge region with a hemisphere and the second,  $1-r^2$  is equivalent to multiplying the edge region with a parabolic dome. Both functions diminish to zero at the edge of the disk.

### Hemispheric Weighting Function

Using a  $\sqrt{1-r^2}$  weighting function, the moment  $H_{pq}$  is given by

$$H_{pq} = \int \int x^p y^q \sqrt{1-(x^2 + y^2)} f(x,y) dx dy \quad (25)$$

The procedure for rotation normalization is unchanged by the weighting function. The rotation normalized moments with respect to the  $x$ -axis over the unit circle may be simplified to

$$H'_{p0} = \frac{\pi}{2} \int x^p (1-x^2) f(x,y) dy dx \quad (26)$$

With some manipulation the edge location may be obtained by

$$\theta = \frac{30H'_{20} - 6H'_{00}}{24H'_{10}} \quad (27)$$

Once  $\theta$  has been determined,  $k$  and  $h$  may be found as follows:

$$k = \frac{8H'_{10}}{\pi(\theta^2 - 1)^2} \quad (28)$$

$$h = \frac{3}{2\pi} H'_{00} - \frac{k}{4} (2 - 3\theta + \theta^3) \quad (29)$$

### Parabolic Weighting Function

Using a  $1-r^2$  weighting function, the moment  $G_{pq}$  is given by

$$G_{pq} = \int \int x^p y^q (1-(x^2 + y^2)) f(x,y) dx dy \quad (30)$$

The procedure for rotation normalization is unchanged by the weighting function. The rotation normalized moments with respect to the  $x$ -axis over the unit circle may be simplified to

$$G'_{p0} = \frac{4}{3} \int x^p \sqrt{1-x^2} f(x,y) dy dx \quad (31)$$

With some manipulation, the edge location  $\theta$  may be obtained by

$$\theta = \frac{6G'_{20} - G'_{00}}{5G'_{10}} \quad (32)$$

Once  $\theta$  has been determined,  $k$  and  $h$  may be obtained as follows:



$$k = \frac{15G_{10}}{4\sqrt{(1-\rho^2)^3}} \quad (33)$$

$$h = \frac{2M_{00}}{\pi} - \frac{k}{2} + \frac{2k}{3\pi}\theta(1-\rho^2)^{3/2} + \frac{k}{\pi}\theta(1-\rho^2)^{1/2} + \frac{k}{\pi}\sin^{-1}(\theta) \quad (34)$$

## VI. Bias Effects Due to Pixel Quantization

The ideal edge that is fit to the data in Section III does not allow for the quantization effects due to finite pixel size in the real data; i.e., the gray value is assumed constant over each pixel in the real data. This introduces a bias error in the calculated edge location. This is demonstrated in Fig. 5 where the difference between the calculated edge location and the actual edge location is plotted. The ideal edge pattern is generated assuming an ideal continuous edge and a square sampling aperture equal to the pixel size. The ideal edge is oriented vertically and its location is varied from -3.5 pixels to +3.5 pixels from the center of a window of diameter 9 pixels. The error is zero when the edge location exactly matches pixel boundaries. Also, the error is zero when the edge is

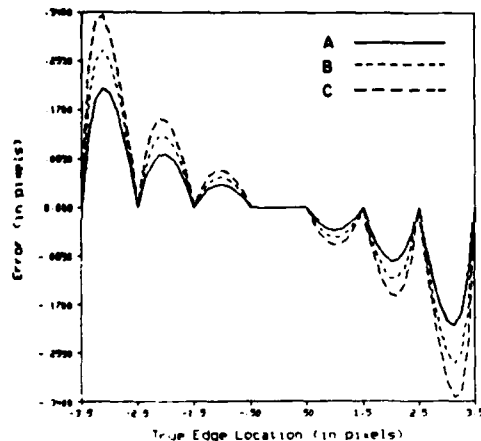


Fig. 5. Bias error effects due to pixel quantization. Shown is the difference between the measured location and the true edge location, and is due to finite pixel size in the moment calculation. The three cases shown are for unweighted (A), hemispheric (B), and parabolic (C) window weightings.

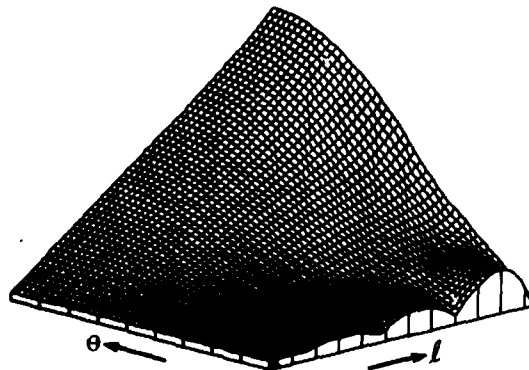


Fig. 6. A three-dimensional plot of the bias error for edge orientations ranging from 0 to 45°, and location ranging from 0 to 3.5 pixels.

located close to the center of the window. When the edge is located at an angle of 45°, the bias error increases considerably. In this case, the edge never is located along pixel boundaries. Shown in Fig. 6 is the bias error for edge locations ranging from 0 to 3.5 pixels from the center of the window, and orientations of the edge ranging from 0° to 45°. Although this bias appears significant, the calculated edge location versus true edge location is always a monotonic function, and thus a table look-up procedure can be used to subtract this bias effect and give perfect edge location results when no noise is present.

## VII. Effects of Noise on Edge Location

Assume additive, independent, identically distributed, Gaussian noise is added to the pixel gray values. The calculated edge location and orientation then become random variables. Assume also that  $\theta$  is zero, consequently the moments need not be rotated. The random variable length can then be viewed as

$$\theta = \frac{4M_{20} - M_{00} + n_1}{3M_{10} + n_2} \quad (35)$$

where (35) is a direct extension of (10). The random variable  $n_1$  is zero-mean and Gaussian with variance

$$\sigma_1^2 = \sigma^2 \iint (4x^2 - 1)^2 dx dy \quad (36)$$

where  $\sigma^2$  is the variance of the additive noise and the integration is defined over the unit circle. The random variable  $n_2$  is also zero-mean and Gaussian with variance

$$\sigma_2^2 = \sigma^2 \iint (3x)^2 dx dy \quad (37)$$

The variances of the random variables arise from the multiplication of the additive noise with the moment masks described in Section IV. Since the noise values are independent, zero-mean, and Gaussian, the density of the summation of  $N$  values remains zero-mean and Gaussian. The multiplicative constants determined by the integrals in (36) and (37) can also be calculated from the moment masks; ie,

$$C_1 = \sum_i \sum_j [4m_{20}(i,j) - m_{00}(i,j)]^2 \quad (38)$$

$$C_2 = \sum_i \sum_j [3m_{10}(i,j)]^2 \quad (39)$$

so that

$$\sigma_1^2 = \sigma^2 C_1 \quad (40)$$

$$\sigma_2^2 = \sigma^2 C_2 \quad (41)$$

where  $m_{20}(i,j)$  is the  $n^{\text{th}}$  moment mask weight for the  $i^{\text{th}}, j^{\text{th}}$  pixel position within the mask window. For a 9 x 9 mask window set, these multiplicative constants are:

Unweighted:	$C_1 = 0.1307$
	$C_2 = 0.3067$
Hemispheric:	$C_1 = 0.0468$
	$C_2 = 0.1082$
Parabolic:	$C_1 = 0.0287$
	$C_2 = 0.0547$

Since these constants are less than unity, the variances of  $n_1$  and  $n_2$  are less than the variance of the additive noise alone.

The numerator and denominator of (35) can now be viewed as the quotient of two independent, non-zero mean Gaussian random variables. The means of the numerator and denominator are simply  $4M_{20} - M_{00}$  and  $3M_{10}$ , respectively, while the variances are that of  $n_1$  and  $n_2$ . Therefore, the length has a Cauchy-like distribution. The Cauchy density has no absolute moments [7]. The

probability density function for the quotient of two independent, non-zero mean Gaussian variables  $X_1$  and  $X_2$

$$W = \frac{X_1/\sigma_1}{X_2/\sigma_2} \quad (42)$$

is given by

$$\begin{aligned} p(w) &= p_1^-(w), & -\infty < w < -1 \\ &= p_1^-(w), & -1 < w < 0 \\ &= p_1^+(w), & 0 < w < 1 \\ &= p_2^+(w), & 1 < w < \infty \end{aligned} \quad (43)$$

where

$$\begin{aligned} p_1^-(w) &= \frac{1}{2\pi} \exp \left[ -\frac{1}{2} \left( \frac{\mu_1^2}{\sigma_1^2} + \frac{\mu_2^2}{\sigma_2^2} \right) \right] \\ &\times \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \sum_{p=0}^{\infty} \frac{1}{(2j)!} \frac{1}{(2k)!} \left( \frac{\mu_1 \sqrt{2}}{\sigma_1} \right)^{2j} \left( \frac{\mu_2 \sqrt{2}}{\sigma_2} \right)^{2k} \\ &\times w^{2(j+k+1)} \frac{\Gamma(j+k+1+1)}{\Gamma(-1)^j} \\ &\pm \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \sum_{p=0}^{\infty} \frac{1}{(2m+1)!} \frac{1}{(2n+1)!} \left( \frac{\mu_1 \sqrt{2}}{\sigma_1} \right)^{2m+1} \left( \frac{\mu_2 \sqrt{2}}{\sigma_2} \right)^{2n+1} \\ &\times w^{2m+2n+p+1} \frac{\Gamma(m+n+p+2)}{\Gamma(-1)^p}, \end{aligned}$$

$$0 < |w| < 1 \quad (44)$$

and

$$\begin{aligned} p_2^-(w) &= \frac{1}{\pi} \exp \left[ -\frac{1}{2} \left( \frac{\mu_1^2}{\sigma_1^2} + \frac{\mu_2^2}{\sigma_2^2} \right) \right] \\ &\times \sum_{j=0}^{\infty} \sum_{k=0}^{\infty} \sum_{p=0}^{\infty} \frac{1}{(2j)!} \frac{1}{(2k)!} \left( \frac{\mu_1 \sqrt{2}}{\sigma_1} \right)^{2j} \left( \frac{\mu_2 \sqrt{2}}{\sigma_2} \right)^{2k} \\ &\times w^{-2(j+k+1)+1} \frac{\Gamma(j+k+1+1)}{\Gamma(-1)^j} \\ &\pm \sum_{m=0}^{\infty} \sum_{n=0}^{\infty} \sum_{p=0}^{\infty} \frac{1}{(2m+1)!} \frac{1}{(2n+1)!} \left( \frac{\mu_1 \sqrt{2}}{\sigma_1} \right)^{2m+1} \left( \frac{\mu_2 \sqrt{2}}{\sigma_2} \right)^{2n+1} \\ &\times w^{-2m-2n-p-3} \frac{\Gamma(m+n+p+2)}{\Gamma(-1)^p}, \end{aligned}$$

$$1 < |w| < \infty \quad (45)$$

An ideal edge oriented vertically and located 1.0 pixels to the right of center is generated and random noise is added to each pixel. The signal-to-noise ratio is defined as

$$SNR = 10 \log_{10} \frac{k^2}{\sigma^2} \text{ dB} \quad (46)$$

where  $k$  is the edge height difference (11) and  $\sigma^2$  is the variance of the additive noise. One thousand independent experiments are performed and the statistics of the measured edge location are recorded. The signal-to-noise ratio is set at 20dB. The sample mean and variance that were recorded are 0.9989 pixels and 0.00885 pixels<sup>2</sup>, respectively. Since the density is Cauchy-like, the moments should not exist in a strict mathematical sense. Therefore, a ranking method is chosen as another statistical measure. The median is used and its value is 0.9986 pixels. Since the sample mean and median are very close in value, the mean is retained as a valid measure.

The variability of  $\theta$  is now considered. Again, zero-mean Gaussian noise is added to the data. The noise case can be considered the extension of (5) such that

$$\theta = \tan^{-1} \left( \frac{M_{01} + n_3}{M_{10} + n_4} \right) \quad (47)$$

The random variables  $n_3$  and  $n_4$  are independent and zero-mean Gaussian. The variances are given by

$$\sigma_3^2 = \sigma^2 \int \int y^2 dy dx = \sigma^2 C_3 \quad (48)$$

$$\sigma_4^2 = \sigma^2 \int \int x^2 dx dy = \sigma^2 C_4 \quad (49)$$

where  $\sigma^2$  is the variance of the additive noise and the limits of integration are the unit circle. The mean values are  $M_{01}$  and  $M_{10}$ , respectively. The multiplicative constants can again be obtained from the moment masks

$$\text{Unweighted: } C_3 = 0.034072$$

$$C_4 = 0.034072$$

$$\text{Hemispheric: } C_3 = 0.012022$$

$$C_4 = 0.012022$$

$$\text{Parabolic: } C_3 = 0.006076$$

$$C_4 = 0.006076$$

The density function of  $\theta$  is similar to that of the length with the transformation

$$z = \tan^{-1} w = \tan^{-1} \frac{x}{y} \quad (50)$$

For an edge location of one pixel, a true value of  $\theta$  equal to 45° and a signal-to-noise ratio of 20dB, 1000 independent experiments were performed. The sample mean was 44.99°, the standard deviation was 1.582°, and the median value was 44.89°. Again, the median and mean value show little difference.

Figs.7-10 represent the mean error and standard deviation of the length for signal-to-noise ratios of 6 dB and 20 dB. The edge locator performs quite well when the edge location is within 2.5 pixels of the center. The RMS error is maximum at the extremes of this region. The RMS error for the 20dB SNR case at 2.5 pixels is only 0.2 pixels. As can be seen in Fig.10, the effects of noise can be reduced by using the radially weighted operators as compared to the unweighted operator. This reduction is greatest near the center of the window. However the weighted operator's performance is reduced when the edge location is not near the center of the window.

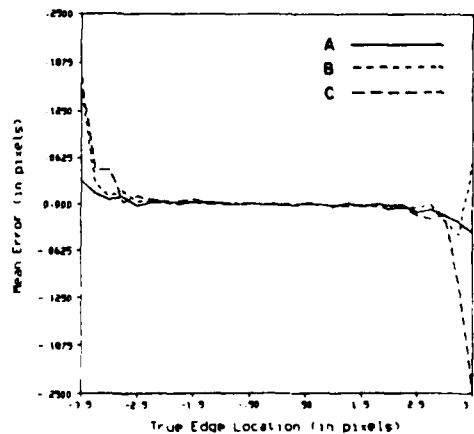


Fig. 7. Sample mean of edge location error versus true edge location. The weightings used are unweighted (A), hemispheric (B), and parabolic (C). The signal-to-noise ratio is 20dB.

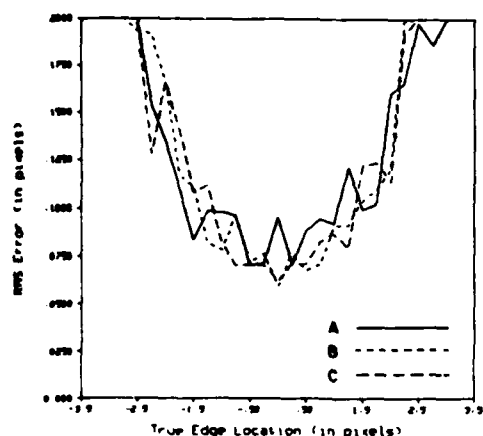


Fig. 8. Sample standard deviation of edge location error versus true edge location. The weightings used are unweighted (A), hemispheric (B), and parabolic (C). The signal-to-noise ratio is 20dB.

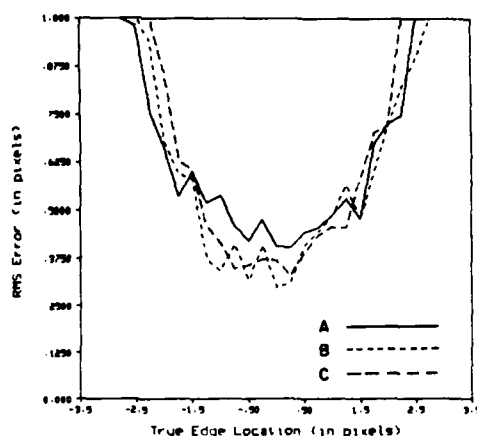


Fig. 10. Sample standard deviation of edge location error versus true edge location. The weightings used are unweighted (A), hemispheric (B), and parabolic (C). The signal-to-noise ratio is 6dB.

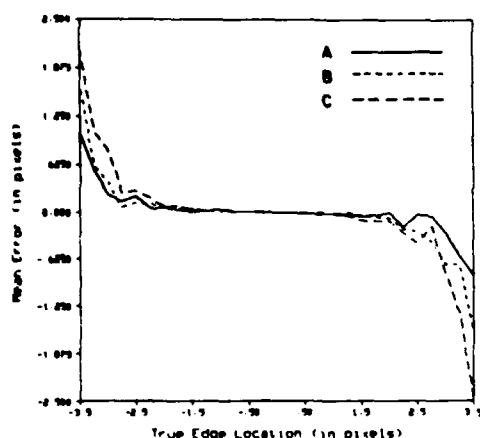


Fig. 9. Sample mean of edge location error versus true edge location. The weightings used are unweighted (A), hemispheric (B), and parabolic (C). The signal-to-noise ratio is 6dB.

### VIII. Conclusions

A new two-dimensional operator has been defined which can accurately locate ideal edges in the presence of noise. The edge orientation and location is calculated by estimating two-dimensional moments directly from the pixel values within the window and requires no iteration. The operator can be used as an edge detector by calculating a confidence measure that indicates how well the ideal edge matches the empirical data.

### References

- [1] A. Rosenfeld and A. Kak, *Digital Picture Processing*, Academic, New York, 1982.
- [2] W. Pratt, *Digital Image Processing*, Wiley, New York, 1978.
- [3] W. Frei and C.C. Chen, "Fast Boundary Detection: A Generalization and a New Algorithm," *IEEE Trans. on Computers*, Vol. C-26, pp. 988-998, October 1977.
- [4] M.F. Hueckel, "An Operator which Locates Edges in Digitized Pictures," *J. Assoc. Comput. Mach.*, Vol. 18, pp. 113-125, 1971.
- [5] M.F. Hueckel, "A Local Operator which Recognizes Edges and Lines," *J. Assoc. Comput. Mach.*, Vol. 20, pp. 634-647, 1973.
- [6] M.F. Hueckel, "Erratum for [5]," *J. Assoc. Comput. Mach.*, Vol. 21, p. 350, April 1974.
- [7] M. D. Springer, *The Algebra of Random Variables*, John Wiley, New York, 1979.

APPENDIX D

DETECTION AND SUB-PIXEL LOCATION OF PHOTOGRAMMETRIC  
TARGETS IN DIGITAL IMAGES

E. M. Mikhail, M. L. Akey, and O. R. Mitchell

presented at the  
Specialist Workshop on Pattern Recognition in Photogrammetry

Graz, Austria

September 27-29, 1983

## **APPENDIX D**

### **DETECTION AND SUB-PIXEL LOCATION OF PHOTOGRAMMETRIC TARGETS IN DIGITAL IMAGES**

**Edward M. Mikhail**  
**School of Civil Engineering**

**and**

**Mark L. Akey and O. R. Mitchell**  
**School of Electrical Engineering**

**Purdue University**  
**West Lafayette, IN 47907, USA**

Data consist of aerial digital images with ground targets in the form of crosses of different dimensions and orientations. Location and recognition of the targets relies on Fourier descriptors and on two-dimensional moments. Further processing employs least squares adjustment of the target shape in order to precisely determine the position (X and Y) and orientation  $\theta$  of each cross to a fraction of a pixel accuracy. Results are given from tests with synthetic crosses on a real terrain digital data base. Accuracies achieved have reached to within 0.03 - 0.05 pixel. Digital image compression has shown to cause cross targets to shift in location by as much as 0.5 pixel.

---

Paper presented at the Specialist Workshop on Pattern Recognition in Photogrammetry.  
Sponsored by the International Society for Photogrammetry and Remote Sensing, Graz,  
Austria, Sept. 27-29, 1983.

## 1. Introduction

Modern photogrammetrists are expected to increasingly deal with digital images. Such images are either directly acquired, as for example by push broom line arrays or multispectral scanners, or indirectly through digitizing photographs. The primary interest has been, and will continue to be, in the extraction of accurate geometric information from the images. This information often concerns well defined features such as edges, lines, crosses, and the like.

Working with digital images allows the photogrammetrists to avail themselves with a variety of digital image processing operations for different purposes. The characteristics of these processing operations are known in general terms to those specializing in image processing. However, to the authors' knowledge, their precise effect on the geometric integrity of the imagery is neither known, nor has it been investigated. Needless to say the photogrammetrists need to know such an effect so that they may be able to properly provide the metric information from the digital images as well as their associated accuracy.

The paper is considered to be a complement to another paper presented by the author at the 39th Photogrammetric Week [7]. The overall objective of the research is the ability, and the accuracy with which, to extract metric information from digital images, and the influence of digital image processing algorithms on the accuracy of such information. The paper in reference [2] covered the chronological development of the research effort. Because of the theme of this Workshop, this paper will concentrate on the automated aspects of the work. We will discuss first the location of edge features, then that of crosses.

## 2. Edge Location using Moment Preserving Method

The method of edge location by moment preserving is described in more detail in [11], and developed at the School of Electrical Engineering, Purdue University.

For simplicity, let us first consider the one-dimensional case, in which an attempt is being made to model a set of data to an ideal step edge as shown in Figure 1. The three parameters defining the edge are:  $h_1$  the signal value below the edge,  $h_2$  the signal value above the edge, and  $X$  the location of the edge. Moment preserving is used as the criterion of best fit of a set  $I$  of  $n$  data points to the ideal edge  $f(s)$ . Rather than solve directly for  $X$ , the edge location is defined as  $k + 1/2$ , where  $k$  is the (unknown) number of samples below the edge. Since there are three unknowns, we set the first three sample moments equal to those associated with the ideal edge, that is:

$$\bar{m}_j = \frac{k}{n} h_1^j + \frac{n-k}{n} h_2^j, \quad \text{for } j = 1, 2, 3 \quad (1)$$

where

$$\bar{m}_j = \frac{1}{n} \sum_{i=1}^n l_i^j \quad (2)$$

is the  $j^{\text{th}}$  sample moment, and  $j$  is a power. The three equations given by (1) may be solved directly in a closed form. In particular, the solution for  $k$  is given by

$$k = \frac{n}{2} \left( 1 - \frac{c}{\sqrt{4 + c^2}} \right) \quad (3)$$

where

$$c = \frac{1}{\sigma^3} [3\bar{m}_1\bar{m}_2 - \bar{m}_3 - 2\bar{m}_1^3]$$

is the skewness of the data, and

$$\sigma^2 = \bar{m}_2 - \bar{m}_1^2$$

From equation (3) it is clear that  $k$  need not be an integer, and therefore sub-pixel edge location is obtained directly. This method of edge location assumes that the data consists of monotonically increasing values. This will not be the case if noise is present. Preprocessing of the data to smooth out noise oscillations improves results significantly. Moment preserving is very simple to apply, and yields unbiased estimates if the edge lies near the center of the area considered. Reduction of any biasing effects are obtained by recentering the area to be modeled with an initial solution.

### 3. Least Squares Location Model

Let  $f(s,t)$  represent the output of a perfect imaging system, that is, the ideal picture function. Consider next a linear, spatially-invariant imaging system with a normalized point-spread function  $p(s,t)$  assumed known. Then let  $\tilde{l}(s,t)$  denote a random variable representing the measurement at sampling position  $(s,t)$ . We may model the measured quantity using the convolution

$$\tilde{l}(s,t) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(\xi,\eta) p(s-\xi, t-\eta) d\xi d\eta \quad (4)$$

Consider now a set of  $u$  parameters  $\mathbf{x}$  which completely characterizes  $f(s,t)$  over the region of interest. Equation (4) may be rewritten as

$$\tilde{l}(s,t) - f(s,t;\mathbf{x}) * p(s,t) = 0 \quad (5)$$

where  $*$  denotes the convolution operation.

Then for the  $ij^{\text{th}}$  picture element which is a sample of  $\tilde{l}(s,t)$  at  $s = s_i$ ,  $t = t_j$ , we may write a linearized condition equation of the form (dropping  $(s,t)$  for simplicity)

$$\tilde{l}_{ij}^0 + v_{ij} + B_{ij} \Delta = -F_{ij}(\mathbf{x}^0) \quad (6)$$

where

$\tilde{l}_{ij}^0$  is the initial estimate for the observation,

$v_{ij}$  is the measurement residual,

$F_{ij}(x) = -f_{ij}(x) * p_{ij}$ ,

$B_{ij}$  is the set of partial derivatives of  $F_{ij}(x)$  with respect to the parameters, evaluated at  $x = x^0$ ,

$x^0$  is the set of initial parameter approximations, and

$\Delta$  is the set of corrections to the parameter approximations.

Equation (6) represents a single condition equation for the model known as Adjustment by Indirect Observations. The total set of equations can then be solved by forming the normal equations in the conventional manner [5].

### One-Dimensional Edge

Consider the ideal model of an edge or discontinuity present in a one-dimensional signal  $f(s)$ , as shown in Figure 1. This may be expressed as

$$f(s) = h_1 + (h_2 - h_1) U(s - X) \quad (7)$$

where  $U$  is the unit step function

$$U(s) = 1, \quad s \geq 0$$

$$= 0, \quad s < 0$$

The one-dimensional form of equation (4) is:

$$\tilde{l}(s) = f(s) * p(s) \quad (8)$$

where  $p(s)$  is the system line-spread function. It can be written in the linearized form of equation (6). If  $p(s)$  is a Gaussian function, the spread edge will take the general form depicted in Figure 2.

Other spread functions, as for example a rectangular function, can be used. It is also possible in the least squares solution to "self-calibrate" by estimating the parameters of the selected spread function. Thus, in addition to  $h_1$ ,  $h_2$ , and  $X$  (see Figure 1), a parameter  $d$ , representing the width of the spread function, is also estimated. Table 1 summarizes some of our early results from both methods.



Data Characteristics			Root Mean Square Error (pixels)	
Spread Width	Spread Type	Noise Level	Moment Preserving	Least Squares
1 pixel	Rectang.	0%	0.073	0.000
		1%	0.074	0.010*
		10%	0.192	0.106*
	Gaussian	0%	0.027	0.003
		1%	0.032	0.015
		10%	0.223	0.164
assumed unknown	Rectang.	0%	0.007	0.000
		1%	0.024	0.023
		10%	0.299	0.231
	Gaussian	0%	0.011	0.000
		1%	0.033	0.033
		10%	0.384	0.346

**Table 1.** Edge pointing with simulated data. The starred (\*) values were obtained using best parameter approximations.

The tests made were limited to simulated data with two types of spread function, and therefore all statements regarding the performance of the algorithms should be interpreted with this in mind.

- 1) The least squares model using a rectangular spread function of known width did not function well in the presence of noise. This was due to the fact that there was no redundancy for the determination of the edge location, and because the form of the condition equations could lead to improper convergence when poor approximations were used.
- 2) The least squares model using a Gaussian spread function of known width, performed well in the presence of noise. There was no instability associated with high levels of noise nor with the use of parameter approximations which were poorly selected.
- 3) The extended least squares model, in which the width of the edge spread was determined, performed well for both types of spread function. It is believed that the case with the rectangular spread function did not exhibit the same instability as the previous case because the number of measurements provided a redundancy for the determination of all parameters. This indicates that the original model would have operated satisfactorily had there been more than one measurement in the spread area of the step.
- 4) The precision of the estimate of edge location is dependent only upon the width of the spread function and the signal-to-noise ratio. It does not appear to be adversely affected when the width of the spread function must also be determined. The adjustment is also relatively insensitive to variations in the position of the

edge within the area being modeled, provided that the edge is at a distance greater than the spread width  $d$  from one extreme of the step.

- 5) Comparison with the method of edge location by moment preservation indicates that for the regular model the least squares fitting provides the better solution. For the extended model, the two methods give comparable results at low levels of signal noise. The most noticeable difference is for the case of perfect data, that is, without added noise, when the least-squares method yields errors only due to round-off. In no case does the method using moment preservation yield smaller errors than by the least squares technique.
- 6) The extended least squares model has the advantages of providing estimates for both edge location and edge spread. However, when compared with the moment preserving method it is computationally less efficient, and also requires initial approximations for all unknowns. The moment preserving method contains a bias when the edge is not located near the center of the area under consideration. This is not believed to be a serious problem in most practical cases. The results indicate that the method of moment preservation offers a reliable solution to the problem, without requiring any assumptions or modeling of the spread function. However, the method of least squares has the potential of providing higher accuracies, particularly if started with good approximations. Investigations with the cross targets to follow substantiate this fact.

#### **4. Investigations with Cross Targets**

The main task of this investigation is composed of two steps:

- (1) The automatic detection and approximate locations of several cross targets in a large image using procedures based on pattern recognition and feature extraction techniques; and
- (2) Precise determination of the position of each cross target center using the least squares algorithm.

Each of these tasks is briefly discussed separately.

##### **4.1. Automatic Detection, Recognition, and Location of Cross Targets**

An algorithm has been developed to detect, recognize, and locate ground cross-targets in digital aerial imagery. The algorithm accomplishes these tasks by extracting three major features from the the ground data. Local grey level maxima which correspond to possible cross targets serve as a detection feature. The Fourier descriptors of the contour of these targets provide recognition of the cross as well as approximate location, orientation, and size of the cross. Finally, the two-dimensional moments determine an accurate location, orientation, size, and grey level for each cross. Additionally, a modified version of the algorithm has been developed which uses only the

Fourier descriptors. Each of these tasks is discussed in the following subsections.

#### 4.1.1. Detection

Detection of local grey level maxima is a relatively fast and simple procedure when working with large digital images. Since the grey level of each cross is greater than that of the cross's surrounding background, the cross can be viewed as a local maximum.

To implement local maxima detection, two processes are needed. First, to insure that true maxima are detected and not those maxima that are attributable to system noise, atmospheric effects, etc., a circular convolutional low pass filter is applied to the data. In general, the filter [4] can be expressed as

$$g(x,y) = \iint f(\alpha,\beta) h(x-\alpha,y-\beta) d\alpha d\beta \quad (9)$$

where  $f(x,y)$  is the original two-dimensional image,  $h(x,y)$  is the filter function, and  $g(x,y)$  is the resulting filtered image. Specifically,

$$h(x,y) = \frac{1}{\pi r^2}, \quad \sqrt{x^2+y^2} \leq r$$

$$= 0, \quad \text{elsewhere} \quad (10)$$

where  $2r$  is the diameter or size of the window. Assuming that the size of the convolving filter window is smaller than smallest expected cross size, the grey level structure of the low pass filtered cross can be viewed as a local maximum in two-dimensions.

Second, given that the two-dimensional local maxima are present, a process must be developed to find the peaks of these maxima. To implement this feature, each point in the image is considered. At each point (called a hub point), the image is observed in each of eight directions extending radially away from the hub point. For each direction, the grey level of each data point in that direction is compared to the hub point. If each grey level is lower than the hub point and if one grey level is lower by a specified amount, and the distance from this point to the hub point is less than a given distance, then the hub point is a local maximum in this direction. All eight directions must be satisfied in this way for a hub point to be considered as a two-dimensional local maximum.

All point locations which are detected as two-dimensional local maxima serve as possible cross locations and only these locations are considered for further analysis.

#### 4.1.2. Recognition

The recognition process accomplishes two tasks. First, the process needs to discriminate buildings, road intersections, and other physical objects from crosses since all these objects may be two-dimensional local maxima. Second, given that the object

is a cross, the process must recognize it's approximate two-dimensional orientation, size, and location.

Recognition is accomplished with the use of Fourier descriptors. To obtain the Fourier descriptors of an object, the image must be grey value thresholded to yield a binary image. If the threshold is chosen correctly, the object will be segmented from the background data. Typically, many thresholds are tried to successfully segment the data. A Fourier transform is applied to the contour or boundary of the segmented object to produce the object's Fourier coefficients. The coefficients (descriptors) are normalized for comparison to the coefficients of a "true" cross. If the descriptors match those of a cross within a specified accuracy, the object is classified as a cross. If the descriptors do not match, another grey level threshold is selected for segmentation until the descriptors match or until all the possible thresholds have been exhausted in which case the object is rejected as a cross.

The boundary function of an object can be expressed as

$$\gamma(t) = x(t) + iy(t) \quad (11)$$

where  $x(t)$  and  $y(t)$  are the  $x$  and  $y$  position of the contour at time  $t$  as shown in Figure 3. The boundary function is complex to provide for changes in both  $x$  and  $y$ , i.e.,  $x$  positions on the real axis and  $y$  positions on the imaginary axis. Also, by tracing around the contour in a counterclockwise direction,  $\gamma(t)$  becomes a function of time. The total time to trace around the contour at a uniform speed is one period  $T$ .

Given that  $\gamma(t)$  is a continuous, bounded, periodic function,  $\gamma(t)$  can be expanded into a Fourier series [3],

$$\gamma(t) = \sum_{n=-\infty}^{\infty} c_n e^{i \frac{2\pi n}{T} t} \quad (12)$$

where

$$c_n = \frac{1}{T} \int_0^T \gamma(t) e^{-i \frac{2\pi n}{T} t} dt \quad (13)$$

A great deal of work has been concentrated on Fourier series expansion [2,8,10]. Stemming from this work are a few fundamental properties. Particularly of use is the ability to translate, rotate, scale, and move trace starting point.

**Translation:** If an object is translated by

$$z_0 = x_0 + iy_0$$

then

$$\gamma'(t) = \gamma(t) + z_0$$

which implies

$$c_0' = c_0 + z_0$$

$$c_n' = c_n, \quad n \neq 0$$

**Rotation:** If an object is rotated about the origin by an angle  $\alpha$ , then the coefficients have a constant phase term added such that

$$\gamma'(t) = \gamma(t) e^{i\alpha}$$

then

$$c_n' = c_n e^{i\alpha}$$

**Scale:** If an object is scaled by a factor  $\lambda$ , then

$$\gamma'(t) = \lambda \gamma(t)$$

$$c_n' = \lambda c_n$$

**Starting Point Shift:** If the starting point of the trace is shifted by  $t_0$ , then

$$\gamma'(t) = \gamma(t + t_0)$$

$$c_n' = c_n e^{i \frac{2\pi n}{T} t_0}$$

**N-fold Rotational Symmetry:** If an object exhibits N-fold rotational symmetry about the origin, then the same trace can be obtained by rotating the object by an integral multiple of the angle  $\alpha = \frac{2\pi}{N}$  and moving the starting point clockwise by  $\frac{kT}{N}$ .

$$\gamma(t) = \gamma(t - \frac{kT}{N}) e^{i \frac{2\pi k}{N}}$$

$$c_n \neq 0, \quad (n-1) \bmod N = 0$$

$$c_n = 0, \quad \text{elsewhere}$$

For discrete images, the contours that are traced are polygons. The polygon consists of linear increments in time and position, and therefore is piece-wise continuous. The Fourier coefficients can be determined from these increments by the discrete Fourier transform (DFT). The coefficients as given by the DFT are

$$c_n = \frac{T}{4\pi^2 n^2} \sum_{p=1}^k \frac{\Delta \gamma_p}{\Delta t_p} \left\{ e^{-i \frac{2\pi n}{T} t_p} - e^{-i \frac{2\pi n}{T} t_{p-1}} \right\}, \quad n \neq 0 \quad (14)$$

where  $t_p = \sum_{i=1}^p \Delta t_i$ ,  $t_0 = 0$ ,  $k$  is the number of sides on the polygon, and  $T = t_k$ . For the

$c_0$  term,

$$c_0 = \frac{1}{T} \sum_{p=1}^k \left\{ \frac{1}{2} \Delta \gamma_p + \gamma_{p-1} \right\} \Delta t_p \quad (15)$$

where  $\gamma_p = \gamma_{p-1} + \Delta \gamma_p$ .

If an eight-directional chain-code is used in representing the discrete contour (Fig. 4), the implementation of the DFT becomes relatively easy. A look-up table can be used and indexed by chain-code.

Due to the four-fold symmetry and the concavity of the cross, many of the Fourier coefficients are zero. In fact, only the -7, -3, 1, 5, and 0 order coefficients have non-zero values among the first 8 harmonics. The zeroth order coefficient relates the position of the center of the cross (center of the contour). The other four coefficients are each spaced by four, verifying four-fold symmetry. Comparison of these coefficients with those extracted from contours of other objects show very little similarity. Few objects show four-fold symmetry and those that do, such as square buildings, are not nearly as concave as the cross. Thus, the Fourier descriptors provide an excellent set of features for discerning crosses from other ground objects.

Additionally, the Fourier descriptors provide information on location, orientation, and size. In fact, the descriptors have to be normalized with respect to location, orientation, and size before direct comparison of the coefficients can be made. As mentioned above, the zeroth coefficient provides the location of the cross.

$$c_0 = \bar{x} + i\bar{y} \quad (16)$$

where  $\bar{x}$  and  $\bar{y}$  correspond to the center of the contour. The combination of the first and minus third coefficient yields the angle of rotation with respect to the x-axis. Determining the angle requires the use of the Fourier series properties. Since the cross has four-fold symmetry, the next largest coefficient is  $c_{-3}$  ( $c_1$  is the largest). To normalize the coefficients, both  $\phi_{c_1}'$  and  $\phi_{c_{-3}}'$  must equal zero, i.e.,

$$c_n' = \frac{c_n}{|c_1|} e^{i(nt_0 + \alpha)} \quad (17)$$

$$\phi_{c_1} = t_0 + \alpha \quad (18)$$

$$\phi_{c_{-3}} = -3t_0 + \alpha \quad (19)$$

Solving for  $t_0$  and  $\alpha$  using Eq. (18) and (19) yields

$$t_0 = \frac{1}{4} (\phi_{c_1} - \phi_{c_{-3}}) \quad (20)$$

$$\alpha = \frac{1}{4}(3\phi_{c_1} + \phi_{c_3}) \quad (21)$$

Thus,  $\alpha$  is the angle of orientation. And, simply the ranges in  $x$  and  $y$  on the contour determine an approximate size of the cross.

Only those locations which are recognized as crosses are passed along for further analysis, along with their corresponding location, orientation, and size.

#### 4.1.3. Location

Two parallel schemes have been developed to accurately locate the crosses as well as determine orientation angle and cross grey level heights. Both schemes use the preprocessed data provided by the detection and recognition routines previously discussed.

#### Location using Moments

In general, the cross' grey level heights ( $h_1$  and  $h_2$ ) are distributed among neighboring pixels according to the location and orientation of the cross. Since the grey values of the pixels hold much of this information, a process which uses the grey levels of the cross as well as the general shape should do well in estimating location and orientation. The two-dimensional grey level moments meet this requirement.

A window of data is extracted from the original image. The location of the center of the window is determined from the Fourier descriptor location results (recognition routine). However, rather than using the typical square window, a cross-shaped window is used. The orientation of the cross-shaped window is determined by the Fourier descriptor orientation result. The cross-shape is amply large enough in width to extract the largest expected cross width. The size of the cross-shaped window is determined by the Fourier descriptor size result.

The window is used solely for noise reduction and does not bias the resulting location and orientation. To insure that no bias is instilled, the average background grey level is subtracted from those grey levels in the window, saturating at zero grey level. Then the window will contain only the cross grey levels with no background grey level present. The two-dimensional moments of the window are calculated. For an image  $f(x,y)$ , the  $(p+q)$ th order moment [9] is given by

$$M_{pq} = \int \int x^p y^q f(x,y) dx dy \quad (22)$$

The normalized first order moment in  $x$  and in  $y$  respectively determine the center of mass of  $f(x,y)$ , i.e.,

$$\bar{x} = \frac{M_{10}}{M_{00}} \quad (23)$$

$$\bar{y} = \frac{M_{01}}{M_{00}} \quad (24)$$

Since the cross is symmetric, this is the final estimate of the cross location. After the moments are translated to this location, rotational moments are used to determine the angle of the cross. To translate the original image  $f(x,y)$  by an amount  $(a,b)$  the following transformation is performed on the original moments

$$M_{pq}' = \sum_{r=0}^p \sum_{s=0}^q \binom{p}{r} \binom{q}{s} a^{p-r} b^{q-s} M_{rs} \quad (25)$$

For example,

$$M_{20}' = a^2 M_{00} + 2a M_{10} + M_{20}$$

Due to four-fold symmetry, the fourth order rotational moments must be used to determine the angle of orientation. Rotational moments are complex and are defined by

$$F_{nl} = \int_{-\pi}^{\pi} \int_0^{\infty} r^{n+1} e^{il\theta} f(x,y) dr d\theta \quad (26)$$

The rotational moments may be obtained from the original moments by the transformation

$$F_{nl} = \sum_{j=0}^{\frac{n-l}{2}} \sum_{k=0}^l (-i)^k \binom{\frac{n-l}{2}}{j} \binom{l}{k} M_{n-l+k-2j, l-k+2j} \quad (27)$$

However, when the image has  $k$ -fold symmetry only the  $F_{kk}$  rotational moment needs to be examined to determine the orientation angle.

$$F_{kk} = \sum_{j=0}^k (-i)^j \binom{k}{j} M_{j, k-j} \quad (28)$$

For four-fold symmetry

$$\begin{aligned} F_{44} &= \sum_{j=0}^4 (-i)^j \binom{4}{j} M_{j, 4-j} \\ &= M_{04} - i4M_{13} - 6M_{22} + i4M_{31} + M_{40} \end{aligned} \quad (29)$$

The angle of orientation is

$$\theta = \frac{\phi_{F_{kk}}}{k} = \frac{\phi_{F_{44}}}{4}$$



$$= \frac{1}{4} \tan^{-1} \left\{ \frac{4(M_{31} - M_{13})}{M_{04} - 6M_{22} + M_{40}} \right\} \quad (30)$$

Finally, the background grey level is determined by the average grey level around the cross, not including the cross, and the grey level of the cross is estimated by the grey level at the center of the cross.

### Location using Fourier Descriptors

This method of determining location is similar to that used in the recognition phase of the algorithm. In that phase, different grey levels are used to threshold and binarize an object at a specified point. The Fourier descriptors of the contours of the binary object are compared with the Fourier descriptors of an ideal cross. If the descriptors matched within a specified error, the preliminary location and orientation determined by the descriptors are passed on to the final location process.

To determine accurate location and orientation of the cross, many grey level thresholds are used. Each grey level threshold produces a contour and therefore an estimate of location and orientation of the cross. Of those thresholds that produce acceptable Fourier descriptor results, only the best fifteen descriptor results are retained. Since the Fourier descriptor error measures the match to an ideal cross, the error may be used as a confidence number. The lower the error, the greater the confidence. This confidence number may then be used as a multiplicative weight with which to multiply the location result. The fifteen best confidence numbers multiplied by their respective locations are summed to give the final estimate of the location. Likewise, the final orientation angle is estimated using these weightings, also.

Each location determined by the Fourier descriptors is a sub-pixel result. However, determining the correct grey level at which to threshold the image is not completely evident by the Fourier descriptor error alone. The best location result from a single contour does not typically occur when the Fourier descriptor error is at a minimum. Fortunately, good, consistent results are obtained near the grey level threshold that results in the minimum Fourier descriptor error as well as the threshold that results in the best location.

Therefore, using an averaged result gives a reasonable good estimate of the location, but not necessarily the best result. Additionally, the averaged estimate is less susceptible to noise variations, and thus results in a more confident answer.

### 4.2. Precise Target Location by Least Squares

The least squares model given in section 3 is extended and applied to the cross target. Figure 5 shows a cross which may be considered to be formed by a set of four rectangular components,  $R_i$ ,  $i=1, \dots, 4$ , each with dimensions  $W$  by  $1/2(L-W)$ . The

parameters to be estimated by the least squares method are:

- (a) Coordinates of its center,  $(X,Y)$ ;
- (b) Orientation angle  $\theta$ ;
- (c) Background and cross grey levels,  $h_1$  and  $h_2$

The length  $L$  and width  $W$ , (Fig. 5), are not directly and simultaneously estimated by least squares at this point. They are calculated separately and input into the algorithm.

The general form, Eq. (5), is linearized to the form in Eq. (6), where  $\Delta$  is a 5 by 1 vector of unknown corrections to approximate values for  $X, Y, \theta, h_1, h_2$ . Because the details of the derivations are rather involved, they and attendant assumptions are not included here and may be found in reference [12].

## 5. Experimentation with Cross Data

Two different sets of data have been used for the various experiments. Each is briefly described next.

### 5.1. Fort Sill Synthetic Images

The digital image files generated for the purpose of measuring the positions of crosses made use of the simulation package SIM previously developed at Purdue University and described by Mikhail et. al. [6]. SIM makes use of an augmented digital data base containing both elevation information and quantized density values from a digitized orthophotograph. This is the source from which imagery may be generated which bears the attributes of an aerial frame photograph, but in a digital form. The data base used contained 1778 rows by 1117 columns each, representing the Fort Sill area of Oklahoma. It was derived from aerial photography flown at a nominal scale of 1:50000, and the spacing between data base elements amounts to 4.8 meters at ground scale. The surface defined represents rolling terrain, with elevation ranging from 350 to 550 meters above sea level.

The program makes use of the collinearity condition as the basis for defining an artificial photo ray which systematically scans the object space. The appropriate image element grey shade is assigned by first determining the intersection of the photo ray and object space surface, and then applying suitable interpolation in grey shade from the four adjacent data base elements. By searching for the surface intersection closest to the camera station, hidden surfaces are effectively removed. The images thus produced simulate the photographic perspective with user-defined interior and exterior orientations, with all inherent displacements due to relief and tilt. As presently written, SIM uses a bilinear interpolation in elevation and in grey shade, but both can be redefined easily.

It is possible to superimpose artificial targets in the terrain model by assigning new grey shade values to specific data base elements. In this way, such targets are included in the image synthesis process, and appear as other natural features in the resultant digital image file. By recording the location of the data base elements modified, the ideal location of the imaged target may be easily determined. This provides a set of ideal image coordinates, which are used to evaluate the errors associated with a given target positioning algorithm. Such an approach has been used in the past in the hard-copy measurement of dot and cross targets, by Unruh and Mikhail [13].

Minor modification of the SIM package was made to permit the generation of several image segments within one program execution, each with the same interior and exterior orientations, but containing only a small portion of the whole image. A single image coordinate system was preserved by the recording of a false origin for each sub-image. Therefore, the location of any feature could be referenced to an overall image system defined by the orientation parameters. This approach was implemented to allow the efficient use of SIM, since it was not at all necessary to generate a large image, but only a set of small images each containing a feature of interest, all referenced to one coordinate system.

In one experiment, a set of nine image files were generated. The exterior orientation was varied, by assigning combinations of three different values of the primary rotation  $\omega$  and three different values of the tertiary rotation  $\kappa$ .

Thus  $\kappa$  took on values  $0^\circ$ ,  $20^\circ$ , and  $45^\circ$ , and  $\omega$  was  $0^\circ$ ,  $5^\circ$ , and  $15^\circ$ . Within one file eight cross targets were imaged. In all cases, the ratio between the average pixel spacing and the data base element spacing was very roughly 1.0. Therefore the approximate dimensions of the crosses in the resultant images were 5 pixel's length by 1 pixel's width.

Table 2 summarizes the results from the first set of experiments. As mentioned previously two types of spread functions were used, the rectangular and Gaussian. The range in root mean square errors in  $\hat{X}$  or  $\hat{Y}$  is from 0.033 to 0.086 pixel, with one case yielding the relatively high value of 0.394.

Image $\kappa/\omega$	With Rectangular Spread			With Gaussian Spread		
	RMSE $\hat{x}$ or $\hat{y}$ (pixels)	RMSE $\hat{\theta}$ (degrees)	RM $\hat{\sigma}_0^2$ (grey levels)	RMSE $\hat{x}$ or $\hat{y}$ (pixels)	RMSE $\hat{\theta}$ (degrees)	RM $\hat{\sigma}_0^2$ (grey levels)
0°/0°	0.051	0.98	13.9	0.086	1.55	13.7
0°/5°	0.052	1.98	13.9	0.041	1.53	13.5
0°/15°	0.062	2.38	14.2	0.054	2.70	13.6
20°/0°	0.065	1.38	14.0	0.066	2.14	13.7
20°/5°	0.041	1.66	13.5	0.048	2.36	13.1
20°/15°	0.045	2.80	13.0	0.048	3.01	12.7
45°/0°	0.033	0.98	12.7	0.040	1.20	12.6
45°/5°	0.394	22.14	16.7	0.390	22.02	16.6
45°/15°	0.041	2.68	13.1	0.038	2.63	12.6

**Table 2.** Cross pointing on imagery with various orientations. Each image contains eight type 1 cross targets of dimensions roughly 5 by 1 pixels.

It can be seen in Table 2 that the low accuracy levels associated with the imagery with  $\kappa$  of 45° and  $\omega$  of 5° are accompanied by large values of root mean square error in  $\hat{\theta}$ . Closer examination revealed that these values are larger than the average due to the poor performance of the pointing algorithms in two instances. In these particular instances, the initial approximations for  $\theta$  were 0°, when in fact the true values should have been close to  $\kappa$  (45°). These poor approximations appear to have allowed convergence of the adjustments to local minima, and resultant residuals in the final estimates were on the order of 45° in orientation and 1.0 pixel in position. The approximations were calculated by a very simple procedure employing cross-correlation with cross templates. The use of pattern recognition and feature extraction algorithms for deriving approximations totally alleviates this problem as shown later on in this paper.

## 5.2. Experiments with the Arizona Test Data

Another test image was obtained by generating cross targets on a digital image using the Arizona test data. This test data was derived from a digitized stereo model formed by two nearly vertical images taken in October 1966 near Guadalupe, Arizona. The cross targets were superimposed on the digitized image.

A 512 x 512 segment of the digital image is used. Twenty-five cross targets were randomly selected and placed on the image. Cross sizes with aspect ratios of 1x7, 1x10, and 1x13 were used. These ratios are considered more practical than the 1x5 ratio used in earlier experiments. The crosses were arbitrarily rotated at various orientation angles. Furthermore, noise was added to the crosses according to a distribution having the same standard deviation as the image background around each cross. It is recognized that this is a severe amount of noise, but we felt that if the algorithms performed reasonably well in this case that we can be confident of the results from other cases with less noise.

Table 3 lists the RMS values calculated from the discrepancies at the 25 crosses for four different cases. Case A is the moment-based method. In case B, only the values of  $X, Y, \theta$  from the moment method are used as approximations in the least square algorithm. In case C, all five values of  $X, Y, \theta, h_1, h_2$  are entered into the least squares algorithm. Finally, case D is the same as case C, except that instead of using fixed values for the cross length  $L$  and width  $W$ , a simple routine is written to estimate these two parameters prior to entering into the least squares algorithm. The last case D gives the lowest RMS values of about 0.05 pixel in  $X$  and 0.03 pixel in  $Y$ , and may therefore be considered as the best that can be expected.

Case		RMS X (pixels)	RMS Y (pixels)
A	Moment-based Method [M]	0.200	0.182
B	Moment & Least Squares (3 param) [M/LS (3)]	0.059	0.057
C	Moment & Least Squares (5 param) [M/LS (5)]	0.050	0.036
D	Moment & Least Squares (5+2 param) [M/LS (5+2)]	0.053	0.029

**Table 3.** RMS values for different cases.

Accepting this level of accuracy in determining the location of crosses in digital images, the next phase of the investigation concerns the effect of image processing operations on the location of such targets. The first image processing operation considered is image compression as discussed in the following section.

## 6. Geometric Effect of Digital Image Compression

The next group of experiments made use of image files over the Arizona test area with 24 superimposed cross targets, as discussed in Section 5.2. Here, however, two different images are used; one in which the cross targets are without any noise (thus simulating reseau marks), and the other with added noise. At each target, the added noise has a more realistic standard deviation which is  $\frac{1}{4}$  of the background standard deviation around the target. For each of these two sets four cases are considered. (See Figure 6 as an example)

1. The original image at 8 bits/pixel
2. An image which has been compressed to 2 bits/pixel
3. An image which has been compressed to 1 bit/pixel
4. An image which has been compressed to 1/2 bit/pixel

In cases 2, 3, and 4, a two-dimensional adaptive cosine transform was used as the compression algorithm [1]. For each of the eight possible image files described above, two different processing procedures were used:

A. An algorithm based on Fourier descriptors and moments is used for detection and location, followed by the least squares algorithm for precise positioning.

B. An algorithm based only on the Fourier descriptors (i.e. without the use of moments) followed by the least squares algorithm.

Therefore, there are 16 cases in total. Unfortunately, we missed one case, and hence the results of only 15 cases are given. We used the letter F to denote Fourier descriptors; M to denote Moments (after Fourier); 8, 2, 1, and 0.5 to represent the compression cases; and the prefix N to indicate cases with noise. A general remark from these cases is that as the number of bits decreases the location of the crosses changes which implies geometric shift. Furthermore, due to significant distortion to some crosses, the algorithm does not recognize them as such and therefore the total number of crosses is reduced. As an example, only 9 crosses out of a total of 24 were found as crosses for the case of 0.5 bit/pixel for noiseless data using the Fourier descriptors.

The results for fifteen cases are summarized in Table 4. Considering the original imagery (8 bit/pixel), the location of a cross can be achieved with an accuracy of 0.03 - 0.05 pixel. Compression to 2 bit/pixel leads to 0.06 - 0.13 pixel; to 1 bit/pixel to 0.16 - 0.18 pixel; and 0.5 bit/pixel to 0.36 - 0.71 pixel.

Method	Case	Number of Targets	Without Least Squares RMS (pixels)			With Least Squares RMS (pixels)		
			X	Y	XY	X	Y	XY
Fourier	Noiseless							
	F8	24	0.080	0.054	0.096	0.027	0.022	0.035
	F2	24	0.095	0.093	0.133	0.049	0.042	0.065
	F1	23	0.200	0.225	0.300	0.144	0.143	0.203
	F.5	9	0.275	0.350	0.445	0.319	0.171	0.362
	Noise							
	NF8	24	0.086	0.104	0.135	0.041	0.033	0.053
	NF2	24	0.141	0.090	0.167	0.058	0.050	0.077
	NF1	23	0.263	0.155	0.305	0.135	0.131	0.188
	NF.5	-	-	-	-	-	-	-
Moment	Noiseless							
	M8	24	0.195	0.259	0.324	0.027	0.020	0.034
	M2	24	0.224	0.299	0.374	0.037	0.037	0.052
	M1	23	0.423	0.482	0.641	0.139	0.103	0.173
	M.5	9	0.572	1.055	1.200	0.322	0.296	0.437
	Noise							
	NM8	24	0.247	0.179	0.305	0.038	0.030	0.048
	NM2	24	0.348	0.272	0.442	0.079	0.112	0.137
	NM1	23	0.403	0.354	0.536	0.113	0.126	0.169
	NM.5	9	0.774	0.815	1.124	0.237	0.670	0.711

**Table 4.** Summary of Fourier and Moment methods.

## 7. Conclusions and Recommendations

- 1) It is possible to locate a cross, with added noise, in a realistic digital aerial image to an accuracy of 0.03 to 0.05 pixel.
- 2) Techniques of pattern recognition and feature extraction are capable of automatically detecting and locating cross targets in digital aerial images.
- 3) The least squares algorithm, following the results from either the Fourier descriptor or Moment algorithms, produces optimum cross positions.
- 4) Digital image compression causes image features, such as crosses to shift in location. The methods devised here can quantify such shifts.
- 5) At low bit/pixel rates, features get distorted and therefore cannot be recognized. This is a serious problem that will need further investigation.
- 6) Other digital image processing operations will be investigated in a manner similar to compression.
- 7) It is hoped that the ability to quantitatively assess the effects of digital image processing operations will lead to further study of the causes for such geometric changes.

### ACKNOWLEDGEMENTS

The work described in this paper has been supported by the U.S. Army Research Office under Contract No. DAAG 29-81-K-0063. The authors express their gratitude to Dr. Steven Mock, the Scientific Program Director. Thanks are also expressed to Dr. J.D. Thurgood of Wild Heerbrugg who worked on the project while a graduate student at Purdue University. The authors would like to also thank J.S. Bethel, D.B. Cantiller, and W.A. Oren of the School of Civil Engineering for assistance with various programming tasks.



## REFERENCES

1. Chen, W.H., and Smith, C.H., "Adaptive Coding of Monochrome and Color Images", IEEE Trans. on Communications, Vol. COM-25, Nov. 1977, pp. 1285-1292.
2. Granlund, G.H., "Fourier Preprocessing for Hand Print Character Recognition", IEEE Trans. Comput., vol. C-21, pp. 195-201, Feb. 1972.
3. Grogan, T.A. and Mitchell, O.R., "Shape Recognition and Description: A Comparative Study". TR-EE 83-22, Purdue University, W. Lafayette, IN. July 1983.
4. McGillem, C.D. and Cooper, G.R., "Continuous and Discrete Signal and System Analysis". New York: Holt, Rinehart, and Winston, Inc., 1974.
5. Mikhail, E.M., "Observations and Least Squares". IEP, New York, 1976; Distributed by University Press of America, 4720 Boston Way, Lanham, MD 20706, USA.
6. Mikhail, E.M., Unruh, J.E., and Alspaugh, D.H., "Image Simulation from Digital Data", Proceedings of the Fall Technical Meeting of the American Society of Photogrammetry, 1977.
7. Mikhail, E.M., "Photogrammetric Target Location to Subpixel Accuracy in Digital Images", 39th Photogrammetric Week, Stuttgart, West Germany, Sept. 19-24, 1983.
8. Persoon, E. and Fu, K.S., "Shape Discrimination using Fourier Descriptors", IEEE Trans. Syst., Man, Cybern. vol. SMC-7, pp. 170-179, Mar. 1977.
9. Reeves, A.P., "The General Theory of Moments for Shape Analysis and the Parallel Implementation of Moment Operations", TR-EE 81-37. Purdue University, W. Lafayette, IN, Oct. 1981.
10. Richard, C.W. and Hemani, H., "Identification of Three-Dimensional Objects using Fourier Descriptors of the Boundary Curve", IEEE Trans. Syst., Man, Cybern. vol. SMC-4, pp. 371-378, July 1974.
11. Tabatabai, A.J. and Mitchell, O.R., "Edge Location to Subpixel Accuracy in Digital Imagery", submitted to IEEE Trans. Pattern Analysis and Machine Intelligence, 1981.
12. Thurgood, J.D., and Mikhail, E.M. "Subpixel Mensuration of Photogrammetric Targets in Digital Images", Technical Report, School of Civil Engineering, Purdue University, No. CE-PH-82-2, August, 1982.
13. Unruh, J.E. and Mikhail, E.M., "Digital Image Simulation for Photogrammetric Applications", Technical Report, School of Civil Engineering, Purdue University,

CE-PH-81-1, 1981.

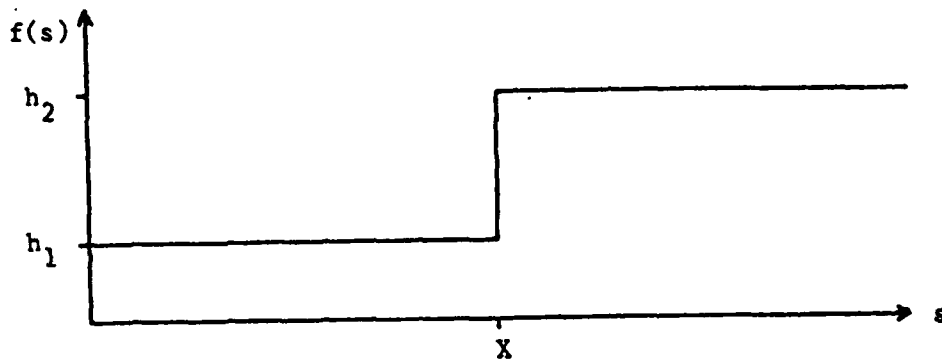


FIGURE 1

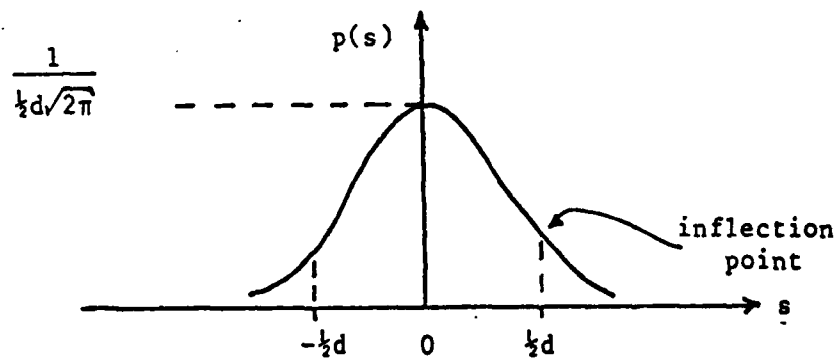


FIGURE 2A

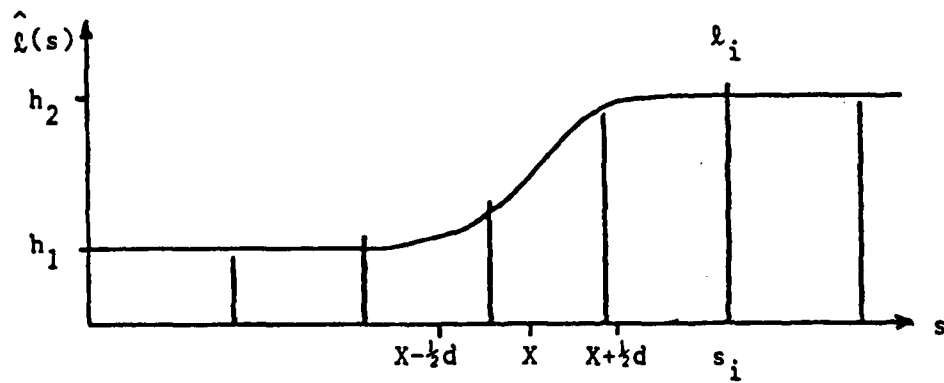


FIGURE 2B

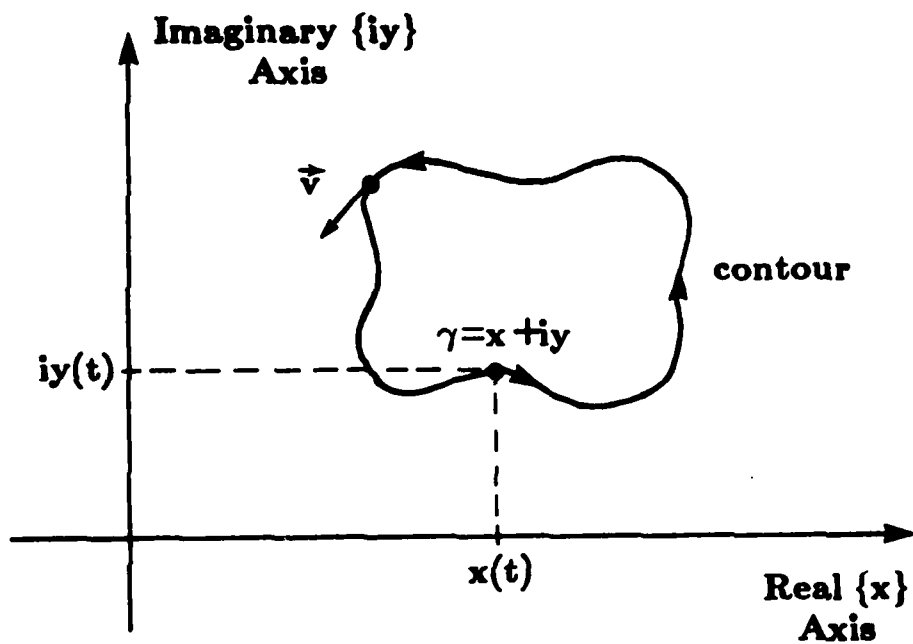


Figure 3. Boundary function in the complex plane.

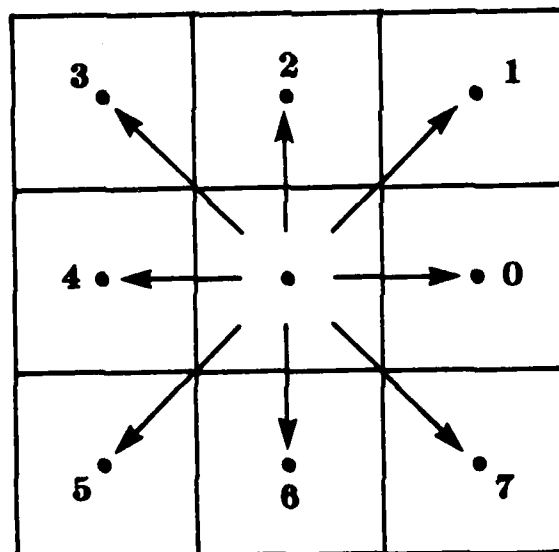


Figure 4. Direction code for chain codes.

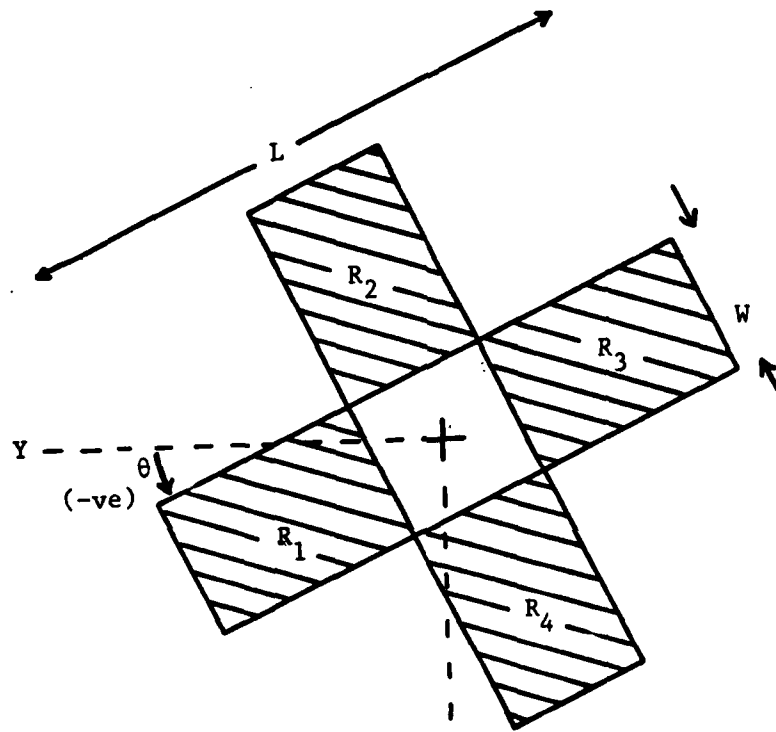


Figure 5. Cross Feature.

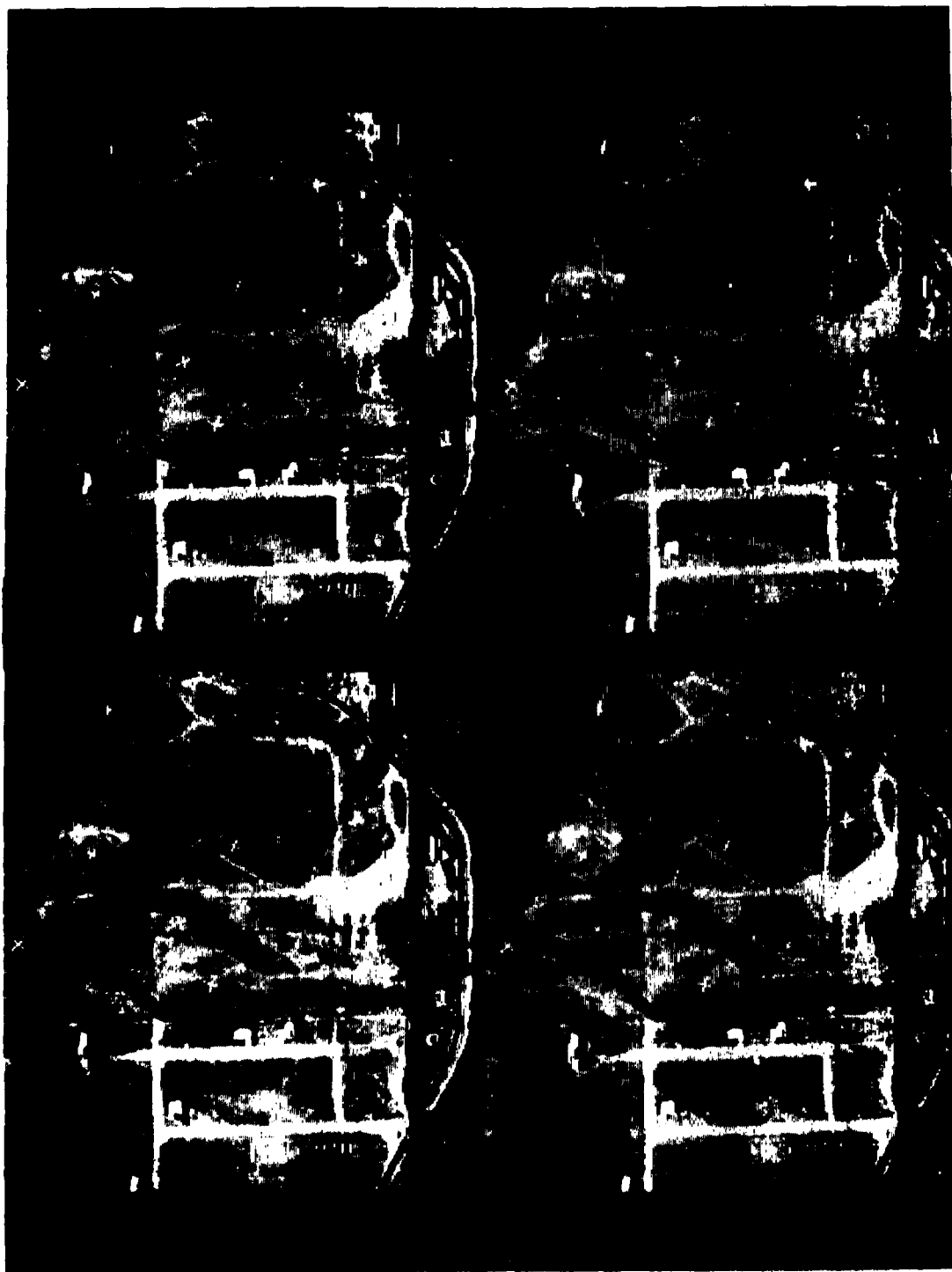


Figure 6. Four sub-images of the Arizona test data. No noise was added to the crosses. (UL) 8 bits/pixel compress, (UR) 2 bits/pixel compress, (LL) 1 bit/pixel compress, and (LR) 0.5 bits/pixel compression.

APPENDIX E

DETECTION AND SUB-PIXEL LOCATION OF OBJECTS IN  
DIGITIZED AERIAL IMAGERY

M. L. Akey and O. R. Mitchell

presented at the  
Seventh International Conference on Pattern Recognition

Montreal, Canada

July 30 - August 2, 1984

## DETECTION AND SUB-PIXEL LOCATION OF OBJECTS IN DIGITIZED AERIAL IMAGERY

Mark L. Akey and O. R. Mitchell  
School of Electrical Engineering  
Purdue University  
West Lafayette, IN 47907, USA

### ABSTRACT

This paper deals with the detection and precision location of ground targets in the form of synthetic crosses on real terrain data. Location of crosses using Fourier descriptors can be achieved to within 0.07 pixels. This method is used to monitor the geometric distortion caused by compressing the data with the cosine transform. Results show geometric "shifts" up to 0.5 pixels may occur after compression. Mean and median filters are applied to synthetic test data. These filters show a minimal amount of geometric "shift" in the presence of noise. The mean filter reduces location error to 0.025 pixels. A new circular two-dimensional median filter is introduced and is shown to instill less geometric distortion than the conventional rectangular median filter.

### 1. Introduction

A long standing concern has been the ability to extract accurate geometric information from digital images. Recent work [1] has shown cross location measurements that are accurate to within one third of a pixel are humanly possible from appropriately digitized images. However, due to inconsistencies between different human observers this margin of accuracy is all but lost. In aerial imagery, the cross is commonly known either as a resseau mark where the cross is directly exposed on the film for registration purposes or as a fiducial mark where the cross is an actual ground feature. The ultimate goal of this work is to study the effects that common image processing techniques have on the metric fidelity of the image. Three different techniques are studied; cosine image compression, mean, and median filtering. In addition, the fidelity of the square two-dimensional median filter is compared to a new circular median filter.

### 2. Automatic Detection, Recognition, and Location of Cross Targets

An algorithm is developed to detect, recognize, and locate ground cross targets in digital aerial imagery. The algorithm accomplishes these tasks by extracting three major features from the ground data. Local grey level maxima which correspond to possible cross targets serve as a detection feature. The Fourier descriptors of the contour of these targets provide recognition of the cross as well as approximate location, orientation, and size of the cross. Finally, the two-dimensional moments determine an accurate location, orientation, size, and grey level for each cross. Additionally, a modified version of the algorithm is developed which uses only the Fourier descriptors in determining the final precision location.

#### 2.1. Detection

Since the grey level of each cross is greater than that of the cross's surrounding background, the cross can be viewed as a local maximum. To implement local maxima detection, two processes are needed. First, to insure that true bright regions are detected and not those maxima that are attributable to noise, a circular convolutional low pass filter is applied to the data.

Second, a process must be developed to find the maxima. To implement this feature, each point in the image is considered. At each point (called a hub point), the image is observed in each of eight directions extending radially away from the hub point. For a given direction, if each grey level is lower than the hub point and if at least one grey level is lower by a significant amount, and if the distance from this point to the hub point is less than a maximum distance, then the hub point is considered a local maximum in this direction. All eight directions must be satisfied in this way for a hub point to be considered as a two-dimensional local maximum.

#### 2.2. Recognition

The recognition process accomplishes two tasks. First, the process needs to discriminate buildings, road intersections, and other physical objects from crosses since all these objects may be two-dimensional local maxima. Second, given that the object is a cross, the process must recognize its approximate two-dimensional orientation, size, and location.

Recognition is accomplished with the use of Fourier descriptors. To obtain the Fourier descriptors of an object, the original unfiltered image must be grey value thresholded to yield a binary image. If the threshold is chosen correctly, the object will be segmented from the background data. Typically, many thresholds are tried to successfully segment the data. A Fourier transform is applied to the contour or boundary of the segmented object to produce the object's Fourier coefficients. The coefficients (descriptors) are normalized for comparison to the coefficients of a "true" cross. If the descriptors match those of a cross within a specified accuracy (typically 90% or higher), the object is classified as a cross. If the descriptors do not match, another grey level threshold is selected for segmentation until the descriptors match or until all the possible thresholds are exhausted in which case the object is rejected as a cross.

The boundary function of an object can be expressed as

$$\gamma(t) = x(t) + iy(t) \quad (1)$$

where  $x(t)$  and  $y(t)$  are the  $x$  and  $y$  position of the contour at time  $t$ . The boundary function is considered to be complex to provide for changes in both  $x$  and  $y$ . The total time to trace around the contour at a uniform speed is one period  $T$ .

Given that  $\gamma(t)$  is a continuous, bounded, periodic function,  $\gamma(t)$  can be expanded into a Fourier series [2],

$$\gamma(t) = \sum_{n=-\infty}^{\infty} c_n e^{j \frac{2\pi n}{T} t} \quad (2)$$

where

$$c_n = \frac{1}{T} \int_0^T \gamma(t) e^{-j \frac{2\pi n}{T} t} dt \quad (3)$$

A great deal of work has been concentrated on Fourier series expansion [3].

For discrete images, the contours that are traced are polygons. Each polygon consists of linear increments in time and position, and therefore is piece-wise continuous. The Fourier coefficients are determined from these increments by the direct Fourier transform (DFT). The coefficients as given by the DFT are

$$c_n = \frac{T}{4\pi^2 n^2} \sum_{p=1}^k \frac{\Delta \gamma_p}{\Delta t_p} \left[ e^{-j \frac{2\pi n}{T} t_p} - e^{-j \frac{2\pi n}{T} t_{p+1}} \right], \quad n \neq 0 \quad (4)$$

where  $t_p = \sum_{i=1}^p \Delta t_i$ ,  $t_0 = 0$ ,  $k$  is the number of sides on the polygon, and  $T = t_k$ . For the  $c_0$  term,

$$c_0 = \frac{1}{T} \sum_{p=1}^k \left[ \frac{1}{2} \Delta \gamma_p + \gamma_{p-1} \right] \Delta t_p \quad (5)$$

where  $\gamma_p = \gamma_{p-1} + \Delta \gamma_p$ .

If an eight-directional chain-code is used in representing the discrete contour, the implementation of the DFT becomes relatively easy. A look-up table can be used and indexed by chain-code.

Due to the four-fold symmetry and the concavity of the cross, many of the Fourier coefficients are zero. In fact, only the -7, -3, 1, 5, and 0 order coefficients have non-zero values among the first 8 harmonics. The zeroth order coefficient relates the position of the center of the contour (cross). Few objects show four-fold symmetry and those that do, such as square buildings, are not nearly as concave as the cross. Thus, the Fourier descriptors provide an excellent set of features for discerning crosses from other ground objects.

Additionally, the Fourier descriptors provide information on orientation and size. Before direct comparison of the Fourier descriptors can be made, the coefficients must be normalized with respect to orientation and size. The combination of the first and minus third coefficient yields the angle of rotation with respect to the x-axis. Determining the angle requires the use of the Fourier series properties. Since the cross is four-fold symmetric, the next largest coefficient is  $c_{-3}$  ( $c_1$  is the largest). To normalize the coefficients, both  $\phi_{c_1}'$  and  $\phi_{c_{-3}}'$  must equal zero, i.e.,

$$c_n' = \frac{c_n}{|c_1|} e^{j(n\alpha_0 + \alpha)} \quad (6)$$

$$\phi_{c_1}' = t_0 + \alpha, \quad \phi_{c_{-3}}' = -3t_0 + \alpha \quad (7)$$

Solving for  $t_0$  and  $\alpha$  using Eq. (18) and (19) yields

$$t_0 = \frac{1}{4} (\phi_{c_1}' - \phi_{c_{-3}}'), \quad \alpha = \frac{1}{4} (3\phi_{c_1}' + \phi_{c_{-3}}') \quad (8)$$

Thus,  $\alpha$  is the angle of orientation. And, simply the ranges in  $x$  and  $y$  on the contour determine an approximate size of the cross.

## 2.3. Location

Two parallel schemes have been developed to accurately locate the crosses as well as determine orientation angle and cross grey level heights. Both schemes use the results provided by the detection and recognition routines previously discussed.

### Location using Moments

In general, the cross' grey level heights ( $b_1$  and  $b_2$ ) are distributed among neighboring pixels according to the location and orientation of the cross. Since the grey values of the pixels hold much of this information, a process which uses the grey levels of the cross as well as the general shape should do well in estimating location and orientation.

A window of data is extracted from the original image. The location of the center and the size of the window is determined from the Fourier descriptor location results (recognition routine). The two-dimensional moments of the window are calculated. For an image  $f(x, y)$ , the  $(p+q)$ th order moment [4] is given by

$$M_{pq} = \iint x^p y^q f(x, y) dx dy \quad (9)$$

The normalized first order moment in  $x$  and in  $y$  respectively determine the center of mass of  $f(x, y)$ , i.e.,

$$\bar{x} = \frac{M_{10}}{M_{00}}, \quad \bar{y} = \frac{M_{01}}{M_{00}} \quad (10)$$

Since the cross is symmetric, this is the final estimate of the cross location. After the moments are translated to this location, rotational moments are used to determine the angle of the cross. To translate the original image  $f(x, y)$  by an amount  $(a, b)$  the following transformation is performed on the original moments

$$M_{pq}' = \sum_{r=0}^p \sum_{s=0}^q \binom{p}{r} \binom{q}{s} a^{p-r} b^{q-s} M_{rs} \quad (11)$$

Due to four-fold symmetry, the fourth-order rotational moments must be used to determine the angle of orientation. The rotational moments [4] can be obtained from the original moments

$$F_{44} = M_{04} - i4M_{13} - 6M_{22} + i4M_{31} + M_{40} \quad (12)$$

The angle of orientation is

$$\theta = \frac{\phi_{F_{44}}}{k} = \frac{1}{4} \tan^{-1} \left\{ \frac{4(M_{31} - M_{13})}{M_{04} - 6M_{22} + M_{40}} \right\} \quad (13)$$

Finally, the background grey level is determined by the average grey level around the cross, not including the cross, and the grey level of the cross is estimated by the grey level at the center of the cross.

### Location using Fourier Descriptors

To determine accurate location and orientation of the cross, many grey level thresholds are used. Each grey level threshold yields a contour and therefore an estimate of the cross's location and orientation. Of those thresholds that produce acceptable Fourier descriptor results, only the best descriptor results are retained. Since the Fourier descriptor error measures the match to an ideal cross, the error may be used as a confidence number. The lower the error, the greater the confidence. This confidence number is used as a multiplicative weight. The location and angle of each contour is weighted by this number and summed to obtain the final location and angle. This process gives surprisingly good results with smaller variance than that obtained from just one Fourier descriptor result.



The technique is remarkably similar to the moment technique in one important way. The grey level moments use a continuum of grey values summed according to the moment basis functions to produce the center of the cross. Likewise, the Fourier descriptors use a continuum of grey level thresholds to produce different center of contour locations. However, only those Fourier descriptors results which match a cross are kept. This selective process removes those contours which are greatly effected by noise. The grey level moments on the other hand cannot be manipulated in this way, and must use all grey values.

### 3. Experiments with Cross Data

Two different sets of data have been used for the various experiments. Each is described briefly as well as the testing applied to each.

#### 3.1. Geometric Effect of Cosine Compression

A test image is obtained by generating cross targets on a digital image. This aerial image is from a rural Arizona area. A cross is generated by integrating over that portion of the pixel which contains any part of the cross. The cross targets are then superimposed on the digitized image.

Twenty-four cross targets are randomly selected and placed on the image. Placement is done at an arbitrary sub-pixel location. Cross sizes with aspect ratios of 1x7, 1x10, and 1x13 are used. The aspect ratio relates the width of one leg of the cross to the length of the cross, i.e., 1 unit to 10 units. The widths range from 1 pixel to 1.5 pixels. The crosses are arbitrarily rotated at various orientation angles. Additionally, zero-mean Gaussian random noise is added to the crosses with a standard deviation similar to the standard deviation of the image background around each cross. Two data sets are created; one where a standard deviation of 25% of the background noise is added to the crosses to simulate fiducial marks and one where no noise is added to simulate reseau marks.

For each of the two Arizona test images, a two-dimensional adaptive cosine transform compression scheme [5] is applied. The resulting images are then reconstructed using 8, 2, 1, and 0.5 bits/pixel. For each set, the grey value moment precision technique as well as the Fourier descriptor precision technique is applied. Table 1 shows the mean error in location of each compressed image for both location methods.

As expected, the greater the compression, the greater the error in location of the cross. It should also be noted that for the case of 0.5 bit/pixel compression, many crosses are not recognized (10 crosses) due to the large amount of distortion to the crosses. Many of these crosses are distorted due to the segmenting of the image into 16 by 16 sub-images, the typical size used in Cosine transform compression.

Table 1 Mean location error in pixels from the grey level moment method and Fourier descriptor method. Each method was applied to the various cosine compressions of the Arizona test data.

Compression (bits/pixel)	No Noise		25% Noise	
	Moments	FD	Moments	FD
8.0	0.255	0.073	0.241	0.107
2.0	0.315	0.107	0.325	0.132
1.0	0.527	0.253	0.458	0.224
0.5	0.980	0.470	0.880	0.438

#### 3.2. Geometric Effects of Mean and Median Filters

To better isolate the effects of processing, new test data is artificially generated with known noise statistics. The base image consists of 49 crosses with the aspect ratio 1x7 oriented at random angles and placed at random sub-pixel locations on a flat field. The width of the legs of each cross is set at 3 pixels, thereby making each cross 21 pixels in length. This larger size is necessary to prevent the median filter from removing large portions of the crosses. Three additional images are created by adding varying amounts of independent zero-mean Gaussian random noise to this base image. The standard deviation of the noise is set at 20%, 40%, and 60% of the center step height (grey value) of the cross.

On each of the above images, eight separate processes are performed. These processes included 3 by 3 mean and median filters and circular mean and median filters with diameter 3. The above processes are repeated using 5 by 5 window, and diameters of 5 pixels. To the authors' knowledge, the circular median filter is yet to be introduced in the literature. For the circular window, those pixels on the boundary are weighted according to the amount of pixel interior to the circle. This weight should signify the percentage of that pixel's grey value. However, rather than summing up the total weighted pixel's grey value as in the mean operation, the grey values along with their respect weights are rank ordered lowest to highest according to the grey value. As the grey values are ascended in the ranked order, a running sum of the area that each grey value represents is kept. When the sum reaches 50% of the total circle area, the current grey value is the resulting median of the circular window.

To the four test images, the Fourier descriptor precision method is applied. This method is preferred over the moment method due to it's superior performance on the Arizona test set. Table 2 shows the precision result for each noise case prior to any processing. The mean value is the average unsigned error in location. These statistics follow a Rayleigh distribution and are completely characterized by the mean. Note, for the no noise case, the method does not give perfect locations.

#### Median Results

Table 2 shows the result after processing the image with both the square and circular median filter of size 3 pixels. It should be noted that the zero noise case results in only a minor shift in location for both cases. Since the median filter is known to remove corners from image features some amount of distortion should be observed. However, due to the symmetry of the cross, the distortion is symmetric over the entire cross. Therefore, no shift in location is apparent from the mean undirected error. For the 20% and 40% noise cases, the median filters improve the location results only slightly. However, the 60% case results in a marked reduction in the error.

Table 2 shows the location results from the square and circular median of size 5. Contrary to the size 3 case, there is a slight increase in location error over that of the non-filtered data. The window size is at times too large for the feature causing the median filter to remove the ends of the cross legs. Only in the highest noise case does the median filter increase the accuracy of the Fourier descriptors, thereby outweighing the effects caused by distortion from the filter. Additionally, the circular median results in less error than the square median. This can be attributed to the additional bias the square median has on orientation.

Table[2] Mean location error in pixels for synthetic test image using Fourier descriptor method. Square and circular median and mean filter results are shown using window sizes of 3 and 5 pixels in the presence of various amounts of noise.

Noise	Before Processing	Median				Mean			
		Square		Circular		Square		Circular	
		3x3	5x5	d=3	d=5	3x3	5x5	d=3	d=5
0%	0.061	0.078	0.091	0.078	0.081	0.026	0.043	0.026	0.033
20%	0.108	0.093	0.114	0.094	0.105	0.087	0.118	0.081	0.124
40%	0.207	0.188	0.255	0.184	0.239	0.184	0.258	0.185	0.250
60%	0.404	0.249	0.359	0.242	0.304	0.220	0.300	0.214	0.279

In general, given that the feature is much larger than the window size of the filter, the median improves the accuracy of the Fourier descriptor location result on noisy data. However, when the feature is not larger the window size, the median filter appears to distort the feature and the accuracy of the Fourier descriptor location result is reduced.

#### Mean Results

Table 2 shows the results of the locations after a square mean and circular mean filter of size 3 pixels are applied to the image. For all cases, the location results of the Fourier descriptors are better than prior to preprocessing. This is not seen in the median case where in only the noisier cases did the location result improve over that of the preprocessed result. Also, there is no discernible difference in location accuracy between the square and circular means.

#### 4. Summary

It is shown that the Fourier descriptors can accurately detect and locate cross targets in realistic terrain images to within 0.08 pixels. Accepting this measure as accurate allows the measurement of distortion caused by other image processing schemes. Cosine compression is shown to move the cross as much as 0.5 pixels when the compression rate is 0.5 bits/pixel. As the bit rate is decreased the error is increased. The mean and median show only slight distortion. The distortion is increased when the filter sizes are of moderate size. However, these two filters improve the Fourier descriptor location accuracy when in the presence of noise. The mean filter improves the location results even when no noise is present. Smoothing the data therefore improves the accuracy of the Fourier descriptors and should be considered as a preprocess to the Fourier analysis.

#### ACKNOWLEDGEMENTS

The work described in this paper has been supported by the U.S. Army Research Office under Contract No. DAAG 29-81-K-0063. The authors express their gratitude to Dr. Steven Mock, the Scientific Program Director.

#### REFERENCES

1. Thurgood, J.D., and Mikhail, E.M. "Subpixel Measurement of Photogrammetric Targets in Digital Images", Technical Report, School of Civil Engineering, Purdue University, No. CE-PH-82-2, August, 1982.
2. Grogan, T.A. and Mitchell, O.R., "Shape Recognition and Description: A Comparative Study". TR-EE 83-22, Purdue University, W. Lafayette, IN. July 1983.
3. Persoon, E. and Fu, K.S., "Shape Discrimination using Fourier Descriptors", IEEE Trans. Syst., Man, Cybern. vol. SMC-7, pp. 170-179, Mar. 1977.
4. Reeves, A.P., "The General Theory of Moments for Shape Analysis and the Parallel Implementation of Moment Operations", TR-EE 81-37. Purdue University, W. Lafayette, IN, Oct. 1981.
5. Chen, W.H. and Smith, C.H., "Adaptive Coding of Monochrome and Color Images", IEEE Transactions of Communications, Vol. COM-25, Nov. 1977, pp. 1285-1292.

AD-A147 211

METRIC ASPECTS OF DIGITAL IMAGES AND DIGITAL IMAGE  
PROCESSING(U) PURDUE UNIV LAFAYETTE IN SCHOOL OF CIVIL  
ENGINEERING E M MIKHAIL ET AL. SEP 84 CE-PH-84-9

2/2

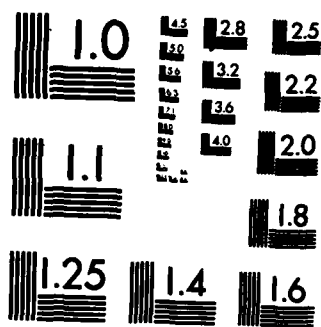
UNCLASSIFIED

ARO-17508. 8-GS DAAG29-81-K-0063

F/G 8/2

NL





## APPENDIX F

### **The Circular Median Filter**

As more research is done in the study of geometric fidelity and how it is affected by common digital processes, it becomes apparent that isotropic filters preserve this fidelity much better than non-isotropic filters. It is thought that circular convolutional windows will produce less distortion in terms of geometric accuracy. Thus, some thought is directed toward the ability to modify existing rectangular window filters.

Due to the rather simple modification of the square mean filter to the circular type, it seems that the modification of the square median filter should also be simple. For the circular window, those pixels on the boundary are weighted according to the amount of pixel interior to the circle. This weight should signify the percentage of that pixel's grey value. However, rather than summing up the total weighted pixel's grey value as in the mean operation, the grey values along with their respective weights are rank ordered lowest to highest according to the grey value. As the grey values are ascended in the ranked order, a running sum of the area that each grey value represents is kept. When the sum reaches 50% of the total circle area, the current grey value is the resulting median of the circular window.

Though the circular median retains its non-linear nature, it has no orientation bias as does the square median. One disadvantage of the square median is that it "lops off" grey level corners in the image. Unfortunately or not, this feature is also present in the circular median. However, the circular median prefers no specific orientation for this to occur. On the other hand, the square median is inconsistent in this response when the corner is not aligned with the axes.

## **APPENDIX G - Participating Scientific Personnel**

1. Mikhail, E.M., Principal Investigator.
2. Mitchell, O.R., Faculty Associate.
3. Tabatabai, A.Y., Received Ph.D. from School of Electrical Engineering, Purdue University in December 1981.
4. Thurgood, J.D., Received Ph.D. from School of Civil Engineering, Purdue University in August 1982.
5. Akey, M.L., Graduate research assistant.
6. Cantiller, D.B., Graduate research assistant.
7. Davis, D., Graduate research assistant.
8. Forstner, W., Visiting Research Scientist.
9. Meenehan, M., Graduate research assistant.
10. Nickols, B., Graduate research assistant.
11. Oren, W.A., Graduate research assistant.
12. Paderes, F., Graduate research assistant.

## **APPENDIX H - Publications, Technical Reports, and Theses**

1. Thurgood, J.D. and Mikhail, E.M., "Photogrammetric Aspects of Digital Images", presented at ASP meeting, Denver, CO. March 14-20, 1982.
2. Tabatabai, A.Y. and Mitchell, O.R., "Edge Location to Subpixel Accuracy in Digital Imagery", IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. PAMI-6, No. 2, March 1984.
3. Tabatabai, A.Y., "Edge Location and Data Compression for Digital Imagery", Ph.D. thesis, December 1981.
4. Thurgood, J.D. and Mikhail, E.M., "Photogrammetric Analysis of Digital Images", Proceedings of Symposium of Commission III, the International Society of Photogrammetry and Remote Sensing, Helsinki, Finland, June 1982.
5. Thurgood, J.D. and Mikhail, E.M., "Subpixel Mensuration of Photogrammetric Targets in Digital Images", CE-PH-82-2, August 1982.
6. Reeves, A.P., Akey, M.L., and Mitchell, O.R., "A Moment Based Two-Dimensional Edge Operator", IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Washington, DC, June 19-23, 1983.
7. Mikhail, E.M., "Photogrammetric Target Location to Subpixel Accuracy in Digital Images", presented at the 39th Photogrammetric Week, Stuttgart, West Germany, Sept. 19-24, 1983.
8. Mikhail, E.M., Akey, M.L., and Mitchell, O.R., "Detection and Sub-pixel Location of Photogrammetric Targets in Digital Images", presented at the Specialist Workshop on Pattern Recognition in Photogrammetry, Graz, Austria, Sept. 27-29, 1983.
9. Akey, M.L. and Mitchell, O.R., "Detection and Sub-pixel Location of Objects in Digitized Aerial Imagery", to be presented at the Seventh International Conference on Pattern Recognition, Montreal, Canada, July 30 - August 2, 1984.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER <b>ARO 17508.8-65</b>	2. GOVT ACCESSION NO. N/A	3. RECIPIENT'S CATALOG NUMBER N/A
4. TITLE (and Subtitle) METRIC ASPECTS OF DIGITAL IMAGES AND DIGITAL IMAGE PROCESSING		5. TYPE OF REPORT & PERIOD COVERED FINAL REPORT April 1981 to September 1984
7. AUTHOR(s) Edward M. Mikhail O. Robert Mitchell		6. PERFORMING ORG. REPORT NUMBER
9. PERFORMING ORGANIZATION NAME AND ADDRESS Schools of Civil and Electrical Engineering Purdue University West Lafayette, Indiana 47907		8. CONTRACT OR GRANT NUMBER(s) DAAG 29-81-K-0063
11. CONTROLLING OFFICE NAME AND ADDRESS U. S. Army Research Office Post Office Box 12211 Research Triangle Park, NC 27709		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS N/A
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		12. REPORT DATE September 1984
		13. NUMBER OF PAGES 98
		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report)  Approved for public release; distribution unlimited.		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)  NA		
18. SUPPLEMENTARY NOTES  The view, opinions, and/or findings contained in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other documentation.		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Photogrammetry      Edge Operators      Image Resampling Digital Image Processing      Fourier Descriptors Metric Accuracy      Image Compression Least Squares Adjustment      Median Filters Moment Preserving      Mean Filters		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) The ability of human pointing in digitized aerial imagery is studied. Automated methods for extracting edge locations in digital imagery are determined with subpixel accuracy. An algorithm for automatically extracting cross targets from aerial imagery is developed and implemented. Crosses are recognized using a method based on Fourier descriptors. Precise locations of the crosses are determined by Fourier descriptors and further refined using a least squares adjustment model. Mean cross location errors are as low as 0.03 pixels.		



UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Metric distortions caused by digital image processing techniques are studied using the Fourier descriptor-least squares algorithms. Cross positions are "shifted" by as much as 0.5 pixels when cosine compression is applied with a 16-fold bit reduction. Mean and median filters show little metric distortions. An isotropic median filter is developed. Resampling with various interpolation functions shows that metric distortion is lower when using bilinear interpolation as compared to nearest neighbor or bicubic interpolation.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

END

FILMED

12-84

DTIC